



Rev[®] Media Management Platform

vbrick rev v7.40

On-Premises Manual Installation Guide - On-Premises



Copyright

© 2021 Vbrick Systems, Inc. All rights reserved.
607 Herndon Parkway, Suite 300
Herndon, VA 20170, USA

This publication contains confidential, proprietary, and trade secret information. No part of this document may be copied, photocopied, reproduced, translated, or reduced to any machine-readable or electronic format without prior written permission from Vbrick Systems, Inc. Information in this document is subject to change without notice and Vbrick assumes no responsibility or liability for any errors or inaccuracies. Vbrick, Vbrick Systems, the Vbrick logo, VEMS Avenger, StreamPlayer, and StreamPlayer Plus are trademarks or registered trademarks of Vbrick Systems, Inc. in the United States and other countries. Windows Media, SharePoint, OCS and Lync are trademarked names of Microsoft Corporation in the United States and other countries. All other products or services mentioned in this document are identified by the trademarks, service marks, or product names as designated by the companies who market those products. Inquiries should be made directly to those companies. This document may also have links to third-party Web pages that are beyond the control of Vbrick. The presence of such links does not imply that Vbrick endorses or recommends the content of any third-party Web pages. Vbrick acknowledges the use of third-party open source software and licenses in some Vbrick products. This freely available source code is posted at <http://www.vbrick.com/opensource>

About Vbrick Systems

Founded in 1998, Vbrick Systems is a privately held company that has enjoyed rapid growth by helping our customers successfully introduce mission critical video applications across their enterprise networks. Since our founding, Vbrick has been setting the standard for quality, performance and innovation in the delivery of live and stored video over IP networks—LANs, WANs and the Internet. With thousands of video appliances installed world-wide, Vbrick is the recognized leader in reliable, high-performance, easy-to-use networked video solutions.

Vbrick is an active participant in the development of industry standards and continues to play an influential role in the Internet Streaming Media Alliance (ISMA), the MPEG Industry Forum, and Internet2. In 1998 Vbrick invented and shipped the world's first MPEG Video Network Appliance designed to provide affordable DVD-quality video across the network. Since then, Vbrick's video solutions have grown to include Video on Demand, Management, Security and Access Control, Scheduling, and Rich Media Integration. Vbrick solutions are successfully supporting a broad variety of applications including distance learning and training, conferencing and remote office communications, security, process monitoring, traffic monitoring, business and news feeds to the desktop, Webcasting, corporate communications, collaboration, command and control, and telemedicine. Vbrick serves customers in education, government, healthcare, and financial services markets among others. Vbrick products are manufactured in an ISO certified manufacturing facility.

Contents

Introduction

What's New in v7.40	1
---------------------------	---

Installation Prerequisites

Server Requirements	1
File Storage	7
Security Scanning Software	7
Network Time Protocol	8
Server Layout and Installed Software	8
Linux Cluster Considerations	9
Configuration Variables	10
Install from Anywhere	13
Installer File Downloads and Configuration	14

Important Version Notes - Before You Start

Rev Analytics	17
Linux: Installation from Anywhere	17
Check Lists	18
Next Steps	19

Using vbrick-setup

Automatic Installation / Upgrade Overview	21
Manual Installation / Upgrade Overview	22
Automatic Installation - Checklist	22
Running the vbrick-setup Installer	23
Next Steps - Post Installer	29

Manual Installation and Upgrades - Getting Started

Manual versus Manual Assist	31
Manual Installation Steps	31
Manual Upgrade Steps	32
Next Steps	33

Manual HAProxy Installation and Upgrade

What to Expect	35
Using vbrick-setup	35
Verifying an HAProxy Installation	36
SSL	36

Next Steps	36
HAProxy Installation Service Specifications	37

Manual Elasticsearch Installation and Upgrade

What to Expect	39
Arbiters	39
Verifying an Elasticsearch Installation	40
Next Steps	42
Elasticsearch Installation Service Specifications	42

Manual MongoDB Installation and Upgrade

What to Expect	43
Arbiters	44
Using vbrick-setup During MongoDB Installation	44
Retrieve and Update the Feature Compatibility Flag	45
Verifying a MongoDB Installation	46
Next Steps	49
MongoDB Installation Service Specifications	49

Rev Windows Components Installation

Windows Service Accounts	52
Install Vbrick Rev Software	52
Configure Windows Firewall	64
Start Vbrick Rev Services	65
Rev Audit Services Configuration - Installation	67
Verify Vbrick Rev Installation	67

Post Installation Review

Configure the Rev Root Account	69
Login as Root User	72
Create a Rev Admin Account	72
Troubleshooting the Installation	73
Elasticsearch Troubleshooting	73
MongoDB Troubleshooting	75
Rev Runtime Service Troubleshooting	77
HAProxy and Load Balancer Troubleshooting	79

Appendices

Appendix A: General Information	81
SSH to a Linux Server Using PuTTY on Windows	81
SSH to a Linux Server Using Terminal and SSH	81
Copy File to a Linux Server Using PuTTY PSCP on Windows	81
Copy a File to Linux Server Using WinSCP on Windows	82

Copy File to a Linux Server Using a UNIX Machine	82
Appendix B: File Storage Path Configuration	82
Rev File Storage	82
Remote File System Considerations	83
Configuring Windows Paths for LocalFolder	83
Change an Existing Storage Path	84
Windows Server 2019	88
Appendix C: Backup Elasticsearch and MongoDB	88
Core Concepts	88
Dependencies	89
Automated Backups	89
Manual Backups	90
MongoDB	91
Appendix D: Load Balancer Information	93
Load Balancer Concepts	93
Example: HAProxy	93
Appendix E: SSL Certificates	96
Private Keys	96
Private Key Length	96
Certificates and Certificate Signing Requests	97
Certificate Generation	97
PEM Files	98
Extras	98
Appendix F: MongoDB Supplemental Information	99
Authentication	99
MongoDB Client	99
MongoDB Replication	99
Red Hat 8.1 Dependencies	100
Appendix G: Operating System Upgrades	101
Windows Server	101
Linux Servers	102
Appendix H: Installation Steps Quick Reference	102
Windows Quick Reference Installation Guide	103
Linux Quick Reference Installation Guide	103

Introduction

This documentation guides you through the steps of the Vbrick Rev on-premises **manual** installation processes. If you want to perform an automated installation or upgrade, you need to view the **On-Premises Automated Installation and Upgrade Guide** version of this documentation. This version can also be found online under the **Automatic Installation** section in Online Help.

What's New in v7.40

- Limited support is provided for older platforms and versions:
 - Versions of Red Hat older than 7.5 are unsupported
- The version of Microsoft dotNet runtime has been updated to 4.8.
- TLS v1.1 has been disabled for new haproxy installations
- New printable quick reference flowchart added: [Appendix H: Installation Steps Quick Reference](#)
-

Installation Prerequisites

Last Updated: June 3, 2021

This document describes the prerequisites required to install Vbrick Rev on a customer's premises.

Caution: Keep in mind that the server configuration specs in this document are the minimum and preferred requirements tested by Vbrick. However, in all cases, server specifications are dependent upon sizing and redundancy requirements. Please refer to the **Hardware Sizing Calculator** available through your solution design or implementation partner when installing a new environment.

Server Requirements

Vbrick Rev On Premises 7.40 provides support for the following Linux distribution and versions:

- Red Hat Enterprise Server 7.8 or 8.2
- Ubuntu 16.04 or 18.04

Note: A Vbrick Rev installation requires both Windows and Linux servers. See below for more details on Windows and Linux requirements

Caution: Vbrick Rev On-Premises 7.34 introduced the optional **Rev Audit Service** feature. If you choose to enable this feature, make sure you consider the impact this could potentially have beyond any capacity you may have already planned. The audit transactions and data can accumulate rapidly and can potentially overwhelm a system that has insufficient storage.

Note: Please note that Rev v7.40 requires DME v3.25 and the new Vbrick Multicast Agent v1.5.

Rev v7.40 is *not* compatible with previous releases of the DME.

Requirement	Description
Vbrick Rev Windows Component Requirements	<p>Vbrick Rev Runtime supports Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, and Windows Server 2019.</p> <p>Note: If your Windows server does not have .Net framework 4.8 or greater, please install or enable .Net 4.8 runtime. More information on .Net framework is found at https://dotnet.microsoft.com/download/dotnet-framework/net48</p> <p>Minimum Configuration Specs:</p> <ul style="list-style-type: none"> • 2 Microsoft Windows Servers • Server 1: 8 GB of RAM / 200 GB of Disk Space • Server 2: 8 GB of RAM / 100 GB of Disk Space, depending upon the number of video files anticipated • Minimum of 8 CPU Cores for all machines. <p>Note: If you are installing the optional Audit service, you may require 3 Microsoft Windows Servers depending upon the audit transactions.</p> <p>Preferred Configuration Specs:</p> <ul style="list-style-type: none"> • 3 Microsoft Windows Servers • 16 GB of RAM • 16 processors configured preferably as 2 virtual sockets, 8 cores per socket <p>Vbrick Rev Node 1: 500 GB (minimum) "Thick Provisioned Eager Zeroed" storage volume, configured as independent and persistent, on a very fast datastore, preferably on a local drive</p> <p>Vbrick Rev Node 2: 100 GB (minimum) "Thick Provisioned Eager Zeroed" storage volume, configured as independent and persistent, on a very fast datastore, preferably on a local drive</p>

General Linux Component Requirements	<ul style="list-style-type: none"> • Either Hardware or VM devices with 64-bit, Ubuntu 16.04 LTS, Ubuntu 18.04 LTS, Red Hat Enterprise 7.5 or higher, or Red Hat Enterprise 8.1 installed and running (see individual requirements for RAM and CPU requirements) • Functional networking (see DNS or static "hosts" file entries) • SSH server and client installed • A <i>universal</i> user that must be the same on all systems. Creating a user account with this name is advised but not necessary. Regardless of the username, that user account <i>must</i> have full sudo access. • Local firewalls (services like <code>firewalld</code> and <code>iptables</code>) should be turned <i>off</i> if possible. Diagnosing firewall issues is outside of the range of this documentation, and if a firewall is active it may cause connectivity issues between the various nodes in the Rev service. Where possible we will provide the ports used by the services we install, but we do not manipulate or configure firewalls with our installer(s). • Broader system security services like selinux and apparmor should be turned <i>off</i> if possible. • A local copy on each system of the Package Installer tarball archive.
Elasticsearch (Linux) Requirements	<p>Runs on Linux server. Elasticsearch is very RAM dependent, the more you can commit to these instances, the better they will perform.</p> <p>Minimum Configuration Specs:</p> <ul style="list-style-type: none"> • 2 Linux machines • Minimum of 100 GB of Disk Space • Minimum of 8 GB of RAM • Minimum of 4 CPU Cores <p>Preferred Configuration Specs:</p> <ul style="list-style-type: none"> • 3 Linux machines • 16 GB or more of RAM • 8 processors. For virtual machines, the ideal configuration is 2 virtual sockets, 4 cores per socket • Virtual machine disks should be provisioned with a "Thick Provisioned Eager Zeroed" storage volume, configured as independent and persistent, on a fast datastore, preferably on a local drive and preferably SSD <p>Note: Elasticsearch data is kept in <code>/var/lib/elasticsearch</code> so keep that in mind for your partitioning schemes. The majority of the available disk space should favor the disk or partition that contains this path.</p>

MongoDB (Linux) Requirements	<p>Runs on Linux server. MongoDB uses both RAM and CPU so do not reduce one in favor of the other, keep both as high as possible.</p> <p>Minimum Configuration Specs:</p> <ul style="list-style-type: none"> • 2 Linux machines • Minimum of 200 GB of Disk Space • Minimum 8 GB of RAM • Minimum of 4 CPU Cores <p>Preferred Configuration Specs:</p> <ul style="list-style-type: none"> • 3 Linux machines • 16 GB of RAM • 8 processors. For virtual machines, the ideal configuration is 2 virtual sockets, 4 cores per socket • Virtual machine disks should be provisioned with a "Thick Provisioned Eager Zeroed" storage volume, configured as independent and persistent, on a fast datastore, preferably on a local drive and preferably SSD <p>Note: MongoDB data is kept in <code>/var/lib/mongodb</code> on Ubuntu systems and <code>/var/lib/mongo</code> on RHEL systems so keep that in mind for your partitioning schemes. The majority of the available disk space should favor the disk or partition that contains this path.</p> <p>Important: MongoDB suggests that the XFS file system be used on the underlying disk/partition that contains this MongoDB data path.</p> <p>Note: If you are installing the optional Audit service, you may require more resources on the Mongo servers depending upon the size of the audit data.</p> <p>RHEL8 Note: MongoDB 4.4 has some additional requirements for Red Hat 7 and 8 that must be satisfied via the licensed Red Hat Repositories. There is additional documentation on the required steps in the Appendix F: MongoDB Supplemental Information section.</p>
------------------------------	---

Load Balancer Requirements	<p>If you have your own load balancers, they can be used in favor of the installer we provide in your production deployment. That said, we still encourage you to install the HAProxy & Auto Setup support to simplify the rest of the installation.</p> <p>Minimum Configuration Specs for CPE Load Balancer:</p> <ul style="list-style-type: none">• A load balancer capable of supporting the following features:<ul style="list-style-type: none">- WebSocket support required- X-Forwarded-For header support required• Minimum of 2 GB of RAM• Minimum of 4 CPU Cores• Please see our Load Balancer FAQ for more information <p>Minimum Configuration Specs for Vbrick Package Installer:</p> <ul style="list-style-type: none">• 1 Linux machine• Minimum 4 GB of RAM• Minimum of 4 CPU Cores <p>Preferred Configuration Specs:</p> <ul style="list-style-type: none">• 1 Linux machine• 4 GB of RAM• 4 processors. For virtual machines, the ideal configuration is 2 virtual sockets, 2 cores per socket• 40 GB "Thick Provisioned Eager Zeroed" storage volume, configured as independent and persistent, on a very fast datastore, preferably on a local drive <p>Note: HAProxy on its own uses very little disk space, however the logging can accumulate under heavy traffic. Make sure you provide sufficient space to <code>/var/log</code> in your partitioning scheme.</p>
----------------------------	---

DNS or Static "hosts" file entries	<ul style="list-style-type: none"> • Create a DNS entry for the Vbrick Rev VIP which is the address used to visit the Vbrick Rev server. • If you are taking advantage of the multi-tenant capabilities of Vbrick Rev, is recommended that you request a wild card DNS entry, for example, *.rev.vbrick.com. In this way, you may provision additional tenants without additional DNS changes using hostnames such as CORPA.rev.vbrick.com and CORPB.rev.vbrick.com. • The Vbrick back-end Elasticsearch and MongoDB servers handle themselves using local "hosts" files so no customization is necessary. <p>Note: If you choose to use DNS, please obtain DNS entries <i>before</i> completing the installation of Vbrick Rev because the original DNS entry will be the only way to connect to the Vbrick Rev Server after initial configuration steps have been completed. If the URL needs to be changed in the future, first have a new DNS entry created then use the Vbrick Rev web interface to update the URL used.</p> <p>Note: Changing Hostname, IP, and MAC address of the Vbrick Rev machine after initial installation is complete requires manual steps, so should be avoided where possible.</p>
HTTPS Certificates	<ul style="list-style-type: none"> • It is recommended to use the Load Balancer as the HTTPS endpoint and connect to Vbrick Rev using regular HTTP. • If you are taking advantage of the multi-tenant capabilities and configuring the Vbrick Rev server to use HTTPS, you must request a wild card HTTPS Certificate entry, for example, *.rev.vbrick.com, so you can provision additional tenants as you can currently only configure Vbrick Rev with one HTTPS Certificate. <p>Note: You may have more flexibility with multiple certificates if you terminate the HTTPS connection on a commercial load balancer rather than on the Vbrick Rev server.</p>
Prerequisite Skills and Knowledge	<ul style="list-style-type: none"> • Comfortable with the Microsoft Windows Server version 2012 and higher • Comfortable running Linux commands • Comfortable configuring virtual machines if installing on Hyper-V or VMware ESXi
Vbrick Resources	<ul style="list-style-type: none"> • Vbrick RevSetup.exe Installer, please refer to the Installer File Downloads and Configuration section. • Vbrick Upgrade and Installation files for Ubuntu and Red Hat, please refer to the Installer File Downloads and Configuration section. • Sample HA Proxy Configuration • Vbrick Rev License File and Serial Number

File Storage

With the exception of HAProxy, all the other devices On-Premises is installed on carry mission-critical data. For this reason there are system and external concerns that need to be addressed and will require due diligence so that a solid plan is in place for both a short and long term plan for where critical data gets stored and how it gets backed up. Backup planning is outside of the scope of this document, we can only suggest that they be performed as routinely as operations permit.

Considerations for Windows

All the content and related files created by users will be stored on a filesystem used by the Windows Rev Runtime server(s). Rev Runtime can make use of any filesystem accessible via drive letter or mounted via UNC path (see [Appendix B: File Storage Path Configuration](#)). The format of the storage can range from simple hardware disks to Network Attached Storage (NAS), to redundant Storage Area Network (SAN) as long as it can be mounted by Windows via SMB.

Multi-node environments require the same network storage path to be mounted on all Rev Runtime servers. Any networked storage should be as redundant as possible and backed up or snapshots taken (as the storage type permits) as regularly as operations can permit. Daily backups may not provide suitable data continuity in the case of recovery in heavily used systems.

Considerations for Linux

Best practice for Linux is to have the data mount location for each service on a separate and exclusive partition. This will allow you to manage size independently from other partitions on the system. Ideally, the data partition should not be shared with anything else that might compete for free space. Please see the specifics for each service for more details on the mount locations for data.

Security Scanning Software

Caution: Security scanning software, such as virus and malware protection, should *not* be run on the Windows servers.
If security scanning software is installed on the Windows server, it must not delete “cookie” files.

In addition, the contents of the following files and directories *must* be left alone. They are required for the system to work.

- C:\Rev (or the directory you installed Vbrick Rev to)
- C:\Program Files\erl9.3
- C:\Program Files\nodejs
- C:\Program Files\RabbitMQ Server
- C:\Windows\.erlang.cookie
- C:\Users\Administrator\.erlang.cookie
- C:\Users\Administrator\AppData\Roaming\npm
- C:\Users\Administrator\AppData\Roaming\RabbitMQ
- C:\Users\USERNAME_RUNNING_REV\.erlang.cookie
- C:\Users\USERNAME_RUNNING_REV\AppData\Roaming\npm

- C:\Users\USERNAME_RUNNING_REV\AppData\Roaming\RabbitMQ

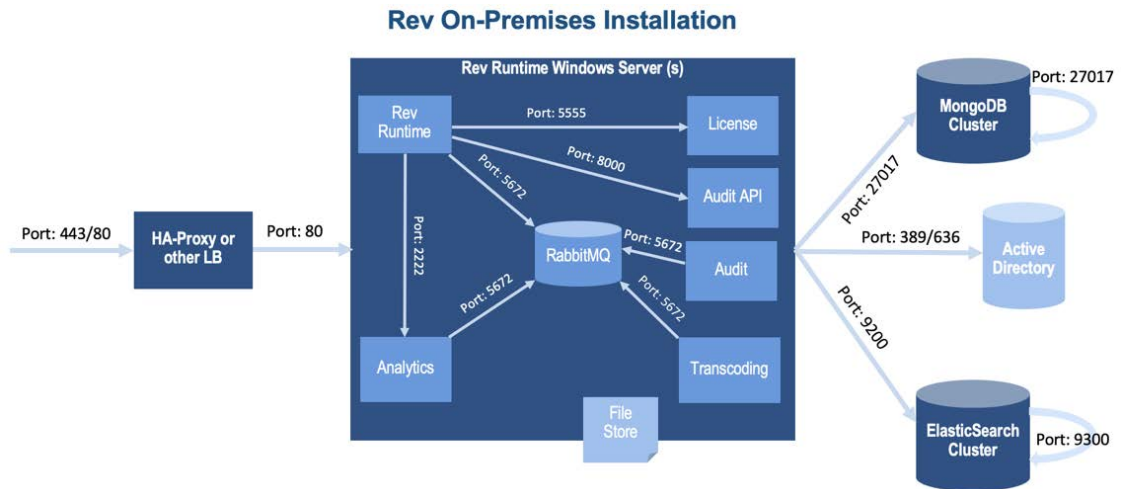
Should files in these directories be altered by the security scanning software, Vbrick Rev must be completely uninstalled and then re-installed to repair the damage.

Network Time Protocol

Although it is not a requirement, we strongly recommend keeping all servers in your Rev network in sync with a network clock. Network Time Protocol (NTP) and the service `ntpd` is used for this and is supported on both Linux and Windows. The NTP clock source is less important than having all your servers having a near-identical system time. The nature of Rev as a system of systems, is that they all rely on each other for data interchange. Differences in clock time can skew the timeliness of a data set so it is critical all your servers are on the same time. Setting up NTP is outside of the scope of this documentation however we cannot over emphasize the critical nature of this service. Many networks offer a central clock source. If one is not available, OEM pools are perfectly acceptable as are the time servers maintained by NIST.

Server Layout and Installed Software

The following software components are installed by the Vbrick Rev installers.



Servers	Version(s)
ElasticSearch Servers	<ul style="list-style-type: none"> • Elasticsearch 7.11.2 • OpenJDK 15.0.1 (packaged with Elasticsearch)
MongoDB Servers	<ul style="list-style-type: none"> • MongoDB 4.4.6
HAProxy Server	<ul style="list-style-type: none"> • Red Hat 7.8: HAProxy 1.7.12 • Red Hat 8.2: HAProxy 1.8.15-6 • Ubuntu 16.04: HAProxy 1.7.12-1 • Ubuntu 18.04: HAProxy 1.8.8

Servers	Version(s)
Rev Runtime Servers	<ul style="list-style-type: none"> • Microsoft .NET Framework 4.8 • Microsoft Visual C++ 2015 Redistributable Package (x64) • Microsoft Visual C++ 2013 Redistributable Package (x64) • Microsoft Visual C++ 2012 Redistributable Package (x64) • Erlang OTP 20 (9.3) • RabbitMQ 3.7.4 • RabbitMQ Presence Exchange Plug-in 3.5.1 • RabbitMQ Stamp Plug-in 1.0.4 • Node.js 14.16.1 • NPM 6.14.12 • Rev Analytics Component • Rev Licensing Component • Rev Transcoding Component <ul style="list-style-type: none"> - FFmpeg 4.2.1 • Rev Runtime Component • Rev Audit Component

Linux Cluster Considerations

Last Updated: June 3, 2021

In deployments of more than one server per service, there are several options for clustering the back-end application data. Both Elasticsearch and MongoDB support $N+1$ redundancy and the end user can take advantage of this with the Rev On-Premises Linux Installer.

Small Deployments

Caution: While Vbrick Rev can support single node installations, we **strongly** suggest that this configuration only be used for development and testing purposes and *not* for Production purposes.

Single Node configurations lack any form of data redundancy and the failure of a single node could result in an unrecoverable data loss.

In our [Installation Prerequisites](#), our minimum deployment allows for one each of both data services (Elasticsearch and MongoDB). For larger deployments involving clusters of servers, even numbers of nodes lack the ability to resolve conflicts so best practice is described as $2n + 1$ - an odd number of nodes, with a minimum of three so that proper quorum can be reached in the cluster. To resolve this issue in a two-node deployment, the Vbrick installer will use the non-primary node in a *service-set* to run an arbiter version of the other service. Generally speaking, Arbiters are non-data carrying cluster members that can act to resolve conflict but cannot act as a point for data recovery in event that node is the last man standing in a disaster. As a result of the Arbiter's reduced duty cycle, it makes an excellent process neighbor to another database.

The `vbrick-vars.txt` file provided with the installer describes a two node Elasticsearch and two node MongoDB deployment, using the B server of each as the arbiter (see [Configuration Variables](#)). The specific configuration rules work as follows:

For arbiters to apply at all, no more than three nodes can be configured in `vbrick-vars.txt` - more than three nodes will disable the feature.

- For Elasticsearch installations:
- If the IP address of Elasticsearch Node 3 matches the IP address of MongoDB Node 2, an Elasticsearch arbiter will be installed on MongoDB Node 2.
- For MongoDB installations: If the IP address of MongoDB Node 3 matches the IP address of Elasticsearch Node 2, a MongoDB arbiter will be installed on Elasticsearch Node 2.

The Pros and Cons of Arbiters

Arbiters serve the extremely critical function of maintaining quorum in a cluster environment, but there's also a very distinct disadvantage: arbiters carry no data. The ramifications of this are highly situational but there are a couple of distinct scenarios you should be aware of. Of the two services, MongoDB is the less critical of the two because the master/slave replication mechanism used by MongoDB keeps a full copy of the data online on each data-bearing member at all times. Under any single node failure, MongoDB will be able to maintain service continuity and operations will continue while the offline server is recovered.

Elasticsearch however, uses an internal mechanism for distributing its indices evenly across all the data-bearing nodes in the cluster. Due to the nature of distributed indices, Elasticsearch clusters require a minimum of two data-bearing nodes to be online for any index operation to succeed. This required minimum means that in Elasticsearch clusters with an arbiter, any node failure that is not the arbiter will result in a non-functional cluster until that node is returned to the cluster. This situation can be especially unfortunate if the failed server and the data on it is lost completely. As a result, an Elasticsearch arbiter is not an ideal mechanism for supporting an Elasticsearch cluster; arbiters continue to work as a solution for limited resources but we strongly suggest you consider your resource allocation and data recovery plans carefully before proceeding down this path.

Large Deployments

If there are no limitations on the number of nodes available for each service, we still suggest three as a good number. Both Elasticsearch and MongoDB support more than three nodes, but for a single organization deployment, the only gains are in redundancy, there is little performance to be gained in an overly large deployment. The configuration details in `vbrick-vars.txt` allow for enumerating additional nodes, and the installers will accommodate these modifications. As long as no two nodes between Elasticsearch and MongoDB share an IP address, no additional (arbiter) steps will be taken beyond the initial service install.

Configuration Variables

Last Updated: June 3, 2021

The core logic of the installer continues to revolve around the configuration file `vbrick-vars.txt`. Please make sure you review this section and understand how Vbrick Setup for Linux is configured as there have been a few changes in the current release.

New in v7.40: You can now test the configuration with the flag `-r` or `--report`. A simple `sudo vbrick-setup -r` tells you very quickly the status of your configuration without having to go through any additional steps.

Note: *Always* double check your work in the `vbrick-vars.txt` file(s). Any mistakes in this file will be carried downstream into the installer's service provisioning and may render the services unusable.

Network interface information required in some previous versions of `vbrick-vars.txt` are no longer used. The following keys and their values can be safely removed from existing configuration files:

- AUTOSETUP
- DNS1
- DNS2
- DOMAIN
- GATEWAY
- INTERFACE
- NETMASK

Automatic Setup

The On-Premises setup process makes every attempt to be as fully automatic as possible. The design and flow are meant to proceed as if the steps are undertaken by an individual installing or updating by hand. Even though the process is adjusted with each new build, every attempt is made to keep the procedure as similar as possible from release to release.

The automatic setup process will import any `vbrick-vars.txt` configuration file present in one of the these two locations, in this order:

- The home directory of the user executing `sudo vbrick-setup` - for example: `/home/vbrick/vbrick.vars.txt`
- Inside the path: `/etc/vbrick`

During the initial setup and extraction process a sample file is copied to `/etc/vbrick/vbrick-vars.txt.sample`. Use this file as a template if an existing configuration file is not available. Successfully imported configurations are renamed as `vbrick-vars.txt.imported`.

Configuration Review

In the initial stage of the installer/upgrade, you'll be given the opportunity to review the configuration graph generated from `vbrick-vars.txt`. This new screen is a change from previous versions, but will hopefully present all aspects of the intended configuration all in one place, without having to go through it line by line. If any changes are required, you can just exit the process with a **ctrl+c** make your changes and then start the process over again.

```
Setup will look for existing configuration files present on the system.
```

```
A sample file has also been copied to '/etc/vbrick'
```

```
(Hit enter to continue.)
```

```
Using vbrick-vars configuration file: /home/vbrick/vbrick-vars.txt
```

```
File '/home/vbrick/vbrick-vars.txt' imported
```

```
On-Premises Configuration Report
```

```
-----
```

```
Multi-node clustering: enabled
```

```
Type: installation
```

```
Files path: /var/lib/vbrick/7.40
```

```
Elasticsearch service:
```

- elastic-a.onpremvbrick.com [10.10.4.124]
- elastic-b.onpremvbrick.com [10.10.4.126]
- elastic-c.onpremvbrick.com [10.10.4.128] *

```
MongoDB service:
```

```
Admin user: admin
```

```
Admin password: *****
```

- mongo-a.onpremvbrick.com [10.10.4.127]
- mongo-b.onpremvbrick.com [10.10.4.128]
- mongo-c.onpremvbrick.com [10.10.4.126] *

```
Rev service:
```

- rev-a.onpremvbrick.com [10.10.4.80]
- rev-b.onpremvbrick.com [10.10.4.81]

```
HAProxy service:
```

- > haproxy.onpremvbrick.com [10.10.4.117]

```
(-> = this node)
```

```
( * = Arbiter)
```

Reference

This is the configuration file distributed in 7.40.

vbrick-vars.txt

```

#
# Default settings for Vbrick Setup.
#
SINGLENODE='n'
SETUPUSER='vbrick'
PASSWORD1='vbrick1'
HAPROXYNODE='10.20.2.210'
HAPROXYHOSTNAME="vbrick-haproxy"
HAPROXYHOSTFQDN="vbrick-haproxy.vbrick.com"
ELASTICNODE1='10.20.2.231'
ELASTICHOSTNAME1="vbrick-elastic-1"
ELASTICHOSTFQDN1="vbrick-elastic-1.vbrick.com"
ELASTICNODE2='10.20.2.232'
ELASTICHOSTNAME2="vbrick-elastic-2"
ELASTICHOSTFQDN2="vbrick-elastic-2.vbrick.com"
ELASTICNODE3='10.20.2.222'
ELASTICHOSTNAME3="vbrick-elastic-3"
ELASTICHOSTFQDN3="vbrick-elastic-3.vbrick.com"
MONGOUSER='admin'
MONGOPASSWD='admin'
MONGONODE1='10.20.2.221'
MONGOHOSTNAME1="vbrick-mongodb-1"
MONGOHOSTFQDN1="vbrick-mongodb-1.vbrick.com"
MONGONODE2='10.20.2.222'
MONGOHOSTNAME2="vbrick-mongodb-2"
MONGOHOSTFQDN2="vbrick-mongodb-2.vbrick.com"
MONGONODE3='10.20.2.232'
MONGOHOSTNAME3="vbrick-mongodb-3"
MONGOHOSTFQDN3="vbrick-mongodb-3.vbrick.com"
REVNODE1='10.20.2.201'
REVHOSTNAME1="vbrick-rev-1"
REVHOSTFQDN1="vbrick-rev-1.vbrick.com"
REVNODE2='10.20.2.202'
REVHOSTNAME2="vbrick-rev-2"
REVHOSTFQDN2="vbrick-rev-2.vbrick.com"
#
# End of default settings for Vbrick Setup.
#

```

It is critical that this file be updated to match the environment the Installer is being deployed in. If the IP addresses don't match, the installers may encounter errors so it's in your best interest to double check the contents of this file before you deploy it.

Install from Anywhere

Rev On-Premises has a feature that allows you to use `vbrick-setup` from any node on the network of a Linux-type supported by On-Premises. The node does not need to participate in the production configuration. In some cases this means you may not require the HAProxy node and this service can be skipped although you must still provide a load balancer of some type in front of Rev.

To disable HAProxy, add this variable to your `vbrick-vars.txt` file *before* you run `vbrick-setup`.

```
HAPROXYINSTALL='false'
```

You should see an acknowledgment of this change during the configuration validation:

```
Rev service:
- qa-w19-rev-12a.lab.vb.loc [10.10.4.209]
- qa-w19-rev-12b.lab.vb.loc [10.20.4.210]

HAProxy service: <skipped>
```

Installer File Downloads and Configuration

Last Updated: June 3, 2021

Both Windows and Linux require the *executable* files distributed by Vbrick as part of each On-Premises release. You will need to ensure these are available on your local file system before you process your desired strategy.

Caution: If you have opted for an automatic installation or upgrade, `vbrick-setup` will handle the ordering in which services are provisioned.

If you have opted for a manual installation or upgrade, please follow the instructions in the sequence they appear. Please do *not* attempt to install more than one node at a time.

File Downloads

Both Windows and Linux have single-file downloads. The bootstrap step to both Installation and Upgrade processes starts with making certain these files are present on your destination systems before you start with any next steps.

Windows

There is a single, standalone installer for Windows Server 2012, 2012 R2, 2016, and 2019. The file name will be a version-numbered name similar to `RevSetup-7.40.XX.YY.ZZ.exe`. In either a fresh installation or upgrade of an existing installation, the Linux servers should be processed first. Make sure you have the Windows installer available on all your intended Windows destinations, and once you've completed the Linux steps, move on to processing the steps documented in [Rev Windows Components Installation](#).

Linux

For Linux, Vbrick distributes a self-extracting installer. The file name will be a version numbered name similar to `vbrick-install-7.40_X.Y.bin`. The self-extracting file is marked executable by default although occasionally these bits are lost due to various forms of Internet file transfer. The installer requires escalated privileges to run; you must use `sudo` to provide this escalation. In our example below we also pro-actively fix permissions on the file we're using, even if it may be an unnecessary step.

```
language bash
chmod 755 vbrick-install-7.40_X.XX.bin
sudo ./vbrick-install-7.40_X.XX.bin
```

The self-extractor deploys all the content necessary for all possible operations and will also upgrade dependencies for that content if necessary.

```
[vbrick@qa-ha-12 ~]$ chmod 755 vbrick-install-7.40_1.8.bin
[vbrick@qa-ha-12 ~]$ sudo ./vbrick-install-7.40_1.8.bin
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing Vbrick On-Premises 7.40 Setup & Installer 100%

Updating Vbrick resources

Inspecting this system

Checking prerequisites
  - Upgrading Chef to 16.6.8

Self-extraction and preparation complete
  - Installation files have been extracted in: /var/lib/vbrick/7.40
  - The setup program is installed in: /usr/sbin
  - You can run a manual setup by typing the command 'vbrick-setup'
  - You can also run an automatic setup by typing the command 'vbrick-setup -a'
```

If you have opted for an automatic installation or upgrade you can delete the file once extraction is complete, it will be of no further use in this process.

Important! If you are performing manual installations or upgrades on your nodes, you will need to copy this file to *each node* where you intend to perform these tasks.

Options

The Linux self-extractor has two options for storing the files and data needed for performing the installation and upgrade tasks. The vbrick-setup installation /upgrade file(s) will detect the change in path automatically, there are no additional changes. The details are in the table below:

Type	Path	Notes
Default	<code>/var/lib/vbrick/7.40</code>	This path is the default, no additional options are necessary.
Custom	Relative to the path the self-extracting installer is run in. Use <code>--keep</code> when running the initial self extraction process and files will be extracted to a directory named <code>vbrick-onprem-7.40_1</code> within the current working directory.	This path will be used on all nodes in the solution. Example: using the <code>--keep</code> option within the vbrick user home directory will yield the path: <code>/home/vbrick/vbrick-onprem-7.40_1</code>

The dynamic file storage path also changes the tests for available space during automatic installation or upgrade. The `vbrick-setup` process uses `df -P` within the working directory to determine available space on the disk or partition that contains the path were the files are kept. The resulting path is displayed both at extraction time and also during configuration confirmation, after running `vbrick-setup`.

```
language bash
```

```
Self-extraction and preparation complete
- Installation files have been extracted in: /home/vbrick/vbrick-onprem-
7.40_1
```

Configuration

Note: If this is a new or first time installation, a sample file is available in `/etc/vbrick/vbrick-vars.txt.sample`. Feel free to use this as a starter template for your configuration.

- If you are performing a **new or first time installation**, edit the sample file or provide your own; save it to `/etc/vbrick/vbrick-vars.txt`.
- If you are **updating an existing installation**, edit your `vbrick-vars.txt`, making sure to include the new and required values documented in [Configuration Variables](#).
- The [Server Layout and Installed Software](#) diagram in the [Installation Prerequisites](#) section illustrates what you are trying to achieve.

The installer handles **Multi-Node** and **Single-Node** installations.

Next Steps

The file that has been self-extracted installs an application named `vbrick-setup` which is capable of handling installations and upgrades in *both* an automated and manual fashion.

For both manual and automatic installations, start with the steps outlined in [Important Version Notes - Before You Start](#).

Important Version Notes - Before You Start

Last Updated: June 3, 2021

The 7.40 On-Premises release has some features that may be of use and might also change the way you choose to tackle your upgrade or installation.

Rev Analytics

Starting with release 7.34, Rev On-Premises includes the ability to enable the Rev Analytics service on your Rev servers. The service itself is small and has little to no impact on the systems where it runs however it has the ability to generate a large amount of data very quickly, depending on how your organization uses Rev. With this in mind if you plan to enable this service, you will want to consider expanding your disk storage plans for MongoDB at the very least. There is unfortunately not a simple calculator for how much storage you might require however there is a way to trim the amount of data kept in Rev Analytics and this may help keep storage within bounds if usage exceeds expectations.

Linux: Installation from Anywhere

Versions previous to 7.40 have had varying support for the launching point for the setup/installer application. In this release we've tried to remove this constraint to the extent possible so that `vbrick-setup` can be run from any of these places:

- Installations can be run from anything on the same network that is one of the supported Linux distributions in this release. It can be any of the machines that will also be participating in the runtime configuration or one that only plays the role of installer.
- Upgrades can be run from anything that ran the 7.28 installation or upgrade.
- Manual Installations continue to be supported.

This feature also allows for HAProxy to be skipped entirely during installation. There is an option that can be added to `vbrick-vars.txt` - there is more information about this optional variable in [Configuration Variables - Install from Anywhere](#).

In instances where `vbrick-setup` has detected that it is playing both the role of the installer/upgrader and as node in the Rev configuration, it will skip some steps because the action being taken is already local. These “skip” steps are normal and there to denote that something was processed even if no actions were taken.

```
Testing each node to make sure we have network connectivity
```

```
Checking Elasticsearch nodes...
```

- 10.10.4.119 [qa-elastic-12a] SSH: ✓
- 10.10.4.120 [qa-elastic-12b] SSH is local: skipped

```
Checking MongoDB nodes...
```

- 10.10.4.121 [qa-mongo-12a] SSH: ✓
- 10.10.4.122 [qa-mongo-12b] SSH: ✓

In the example above we've launched an installation from qa-elastic-12a and the SSH test is skipped because we already have access. In a similar way other situations where access is already established or some content item is already on the file system we skip it and try to notify if we would have normally done this to a remote of the same type.

```
Vbrick Setup will now make sure it has the access necessary to install the systems discovered during the configuration phase.
```

- [qa-elastic-12a] SSH key access: ✓
- [qa-elastic-12b] access is local: skipped
- [qa-mongo-12a] SSH key access: ✓
- [qa-mongo-12b] SSH key access: ✓

```
Does the setup user [vbrick] have full sudo access on the other nodes in this solution? (yes or no)
```

```
yes
```

- [qa-elastic-12a] sudo access: ✓
- [qa-elastic-12b] local: skipped
- [qa-mongo-12a] sudo access: ✓
- [qa-mongo-12b] sudo access: ✓

```
Hit enter to continue.
```

This pattern repeats as often as it is relevant. The skipped status does not carry a negative connotation in this context.

Check Lists

Linux: Installation (All)

- The Linux `vbrick-setup` process relies entirely on the accuracy of the data in `vbrick-vars.txt`. Please ensure you've reviewed the section: [Configuration Variables](#) before you continue with `vbrick-setup`.
- Be cautious when you use copy and paste for code or shell snippets in terminals. Several terminal programs and text editors are capable of transferring special characters via the paste buffer. Some of these characters are hard to distinguish at the destination; characters like “Smart” vs “Dumb” quotes or en-dashes (- vs –) are not always as visually obvious as o vs ö in a terminal. These special characters are usually accomplished via Unicode characters and while they look similar to their simpler counterparts, will not work the same way in shell programs.

Linux: Upgrades (All)

- Always perform as near a full backup as possible on all servers that are part of your Vbrick Rev On-Premises solution with special focus on the Elasticsearch and MongoDB

machines. While the Linux installer supports a backup process during upgrade, we trust the end user also has a process.

- Update your SSL private key size to at least 2048 prior to upgrading.

Linux: Upgrades (Automatic)

- If you are upgrading a system from 7.34, the `vbrick-vars.txt` file was renamed to `vbrick-vars.txt.imported` - there is no need to reuse this file for this upgrade, you can leave the backup file in place.

Linux: Upgrades (Manual)

- Pre-upgrade: the Rev indices in Elasticsearch must be upgraded
- Post-upgrade: the `rev_analytics` database user must be added (optional).
- Post-upgrade: the compatibility version flag must be set.
- Each of these steps is documented here: **Manual Installation and Upgrade Prerequisites**

Windows: Upgrades

- Always check the full system health of all Rev-related services. If you have a multi-node environment, the Elasticsearch and MongoDB clusters must be 100% healthy. Resolve any issues with Elasticsearch or MongoDB before proceeding with any type of upgrade.

Next Steps

Once you have completed all version-specific pre-installation checks, begin your installation type:

- Start at [Using vbrick-setup](#) to perform an automatic installation.
- Start at [Manual Installation and Upgrade Prerequisites](#) to proceed with a manual installation.



Using vbrick-setup

Last Updated: June 4, 2021

The Vbrick installer is a single *application* named `vbrick-setup` and acts as the coordinator for *both* installation and upgrade tasks.

Important Notes

Note: All commands related to the Linux installation or upgrade must be executed with escalated privileges using `sudo`. You must use `sudo` (rather than a fully privileged account) to prefix your commands: `sudo vbrick-setup`.

Caution: Be cautious when you copy/paste code or shell snippets in terminals. Several terminal programs and text editors are capable of transferring special characters via the paste buffer. Some of these characters are hard to distinguish at the destination; characters like “Smart” vs “Dumb” quotes or en- dashes (- vs –) are not always as visually obvious as o vs ö in a terminal. These special characters are usually accomplished via Unicode characters and while they look similar to their simpler counterparts, will not work the same way in shell programs.

Caution: *All* systems intended for upgrade are tested by `vbrick-setup` to ensure the system is in a state suitable for upgrade. These qualification tests are performed in *both* automatic *and* manual upgrades. If the tests fail, use the test report to remedy the issue and attempt the upgrade again. Vbrick On-Premises releases no longer contain OS updates.

Automatic Installation / Upgrade Overview

Automatic installations and upgrades are a fully coordinated process that deploys contents to all the nodes in the Linux solution, installs and configures the corresponding services and confirms the services are in a state ready for use by Rev.

New in this release is the ability to launch the automatic process any network-member machine that matches one of the supported Linux types. This machine does not need to be a participant in the production configuration but it can also be a member, capable of being launched from HAProxy or any of the Elasticsearch or MongoDB nodes. There is no longer a requirement for an 'installation device' although you should continue to use this device if it was created for the previous release.

Rev Runtime requires some form of load balancer to work as a front-end service, even in a single-node environment. If you choose to use the HAProxy node only as a setup server, any

other load balancer service or appliance that meets specifications (see [Installation Prerequisites](#)) will serve as a suitable substitute.

The automatic process is invoked with the `-a` or `-automatic`.

Run vbrick-setup in automatic mode

```
sudo vbrick-setup -a
```

The automatic process goes through a number of steps to make sure it can complete its processes from beginning to end. The tests include `ssh` and `sudo` access in addition to checking for available disk space. These checks are not performed during a manual installation.

To begin an automatic installation or upgrade, jump to [Automatic Installation - Checklist](#).

Manual Installation / Upgrade Overview

Manual installations and upgrades are covered in their own section. Start here: [Manual Installations and Upgrade Prerequisites](#).

Automatic Installation - Checklist

The automated `vbrick-setup` process has several distinct stages of operation and the process will stop and wait for input before proceeding to the next stage. The process and steps are nearly identical for both installation and upgrade and only diverge when it comes time for that specific distinction; all other aspects - content copying, network and user access, and so forth, have identical requirements. Each step is detailed in the table below.

Configuration Validation	The setup process will examine the configuration file and determine the various roles in the deployment. This step is the same for <i>both</i> automatic and manual installations.
Verify Action Type	You will be prompted to confirm the type of action vbrick-setup will be taking: installation or upgrade .
Network and System Access	The setup process will confirm its ability to gain access to the remote systems (including sudo) before trying to continue. If access fails on any point or on any device, the process will exit.
Upgrade Qualification Testing	If the setup process is handling an upgrade, it will test each node to make sure it qualifies for the upgrade.
System Preparation and Bootstrap	The setup process will move files and packages onto the systems detailed in the configuration. In addition, the setup process will prepare all setup/upgrade orchestration to ensure the systems are fully prepared to install or upgrade On Premises components. If an upgrade is being performed, the destination system will be tested to ensure it qualifies for the upgrade.

<u>System Installation/Upgrade</u>	The setup process will execute the orchestration detailed for each system, running to completion
<u>System Status</u>	Post installation/upgrade the setup process will also test its results to make sure it is clear to proceed with other On Premises upgrades.

Note: The images in this document may indicate either installation or upgrade. The first three setup steps in the process are virtually identical. `vbrick-setup` simply substitutes the correct phrase into many sayings as it applies.

Further, if you run the automatic installer, there are no additional steps for Linux when you are done. The Linux installer (`vbrick-setup`) is capable of managing the entire process and if the status at the end is suitable, you can then safely proceed to processing your Windows servers immediately.

Caution: If you intend to follow the upgrade path, make certain all Rev services are *fully stopped* before you begin the upgrade of the Linux components.

`vbrick-setup` will enforce this requirement by checking the listening ports on all the Rev servers listed in your configuration. If any of them are still online, `vbrick-setup` will pause and loop until this condition is changed.

A portion of this upgrade includes an upgrade to the structure of Elasticsearch indices and the output product is not compatible with the running version of Rev. In addition, there is a danger that Rev itself will try to alter these change so it is *imperative* that Rev remain stopped until you've completed the upgrade of not only the Linux components but the Windows components as well.

Running the vbrick-setup Installer

If you've completed the steps in [Installer File Downloads and Configuration](#) and also edited your `vbrick-vars.txt` file, you're ready to proceed. Use the `-a` or `--automatic` flag to enable automatic installations or upgrades.

Run vbrick-setup in automatic mode

```
sudo vbrick-setup -a
```

Remember, the **manual mode** is the **default** for `vbrick-setup` and requires no additional arguments. The manual installation process will perform the same configuration analysis and then install the components that match for that instance.

Run a vbrick-setup manual upgrade

```
sudo vbrick-setup -u
```

If you want to proceed with a manual installation or upgrade, please see the [Manual Installation / Upgrade Overview](#) section and proceed with those steps.

Note: Yes/No questions can be answered with either the entire word or just a simple `y | n`. However, *both* require the `<Enter>` key to complete the data entry

Configuration Validation

```
vbrick@qaha.lab.vb.loc:~$ sudo vbrick-setup -a
Welcome to Vbrick On-Premises automatic setup.

Setup will look for existing configuration files present on the system.

A sample file has also been copied to '/etc/vbrick'

Hit enter to continue. █
```

The critical nature of the configuration used in this process has been emphasized in [Configuration Variables](#). If the configuration file is parsed correctly, the installer will move on to the validation stage where the status screen is presented. The most critical components are listed at the top and emphasized below.

In the examples below we show a Multi-node Upgrade and a Single-node Installation.

- **This node:** should always indicate the machine/device the vbrick-setup is running on.
- **Multi-node clustering:** (values: *enabled/disabled*) this indicates the status of the SINGLENODE flag in `vbrick-vars.txt`
- **Orchestration type:** (values: *installation/upgrade*) this value is determined automatically and is displayed to confirm everything is on the intended path.

If you are pleased with the configuration, hit **Enter** to continue. If you want to make changes use **ctrl+c** to break/end the process, make your edits, and start again.

Verify Action Type

After `vbrick-setup` displays the configuration, it will request confirmation of the type of action it needs to undertake.

```
You have chosen type: installation

Please confirm your choice by entering the whole word below
Type: installation
Hit enter to continue.
```

You must type the *whole word*, or you will be asked to make the entry again. The process will exit if this confirmation fails three times.

```
You have chosen type: upgrade

Please confirm your choice by entering the whole word below
Type: upgrade
Hit enter to continue.
```

You can use **ctrl+c** at any time to exit this process.

Network and System Access

Once the configuration is confirmed, the setup process moves on to confirming it is able to reach everything in the solution via SSH.


```

Testing each node to make sure we have network connectivity

Checking Elasticsearch nodes...
- qaes01 [10.150.2.12] SSH: ✓
- qaes02 [10.150.2.13] SSH: ✓

Checking MongoDB nodes...
- qamongo01 [10.150.2.14] SSH: ✓
- qamongo02 [10.150.2.15] SSH: ✓

Hit enter to continue. █

```

On the next step, Vbrick will confirm the setup user specified in the configuration has the access necessary to proceed with an installation or upgrade. If necessary, `vbrick-setup` will create SSH keys and set up keyed access on each device in the solution.

```

Vbrick Setup will now make sure it has the access necessary to upgrade the
systems discovered during the configuration phase.

- [qaes01] SSH key access: ✓
- [qaes02] SSH key access: ✓
- [qamongo01] SSH key access: ✓
- [qamongo02] SSH key access: ✓

Does the setup user [vbrick] have full sudo access on the other nodes
in this solution? (yes or no)
yes

- [qaes01] sudo access: ✓
- [qaes02] sudo access: ✓
- [qamongo01] sudo access: ✓
- [qamongo02] sudo access: ✓

Hit enter to continue. █

```

Upgrade Qualification Testing

At this point, *if* an upgrade is being processed, each node will be tested in turn to make sure it qualifies for the intended upgrade. The test controls as applied to each node are relatively simple:

- Node *must* be Ubuntu 16.04, Ubuntu 18.04, Red Hat 7.5, or Red Hat 8.1
- Node *must* be running one or more of the following software packages:
 - Elasticsearch 6.3.x where x is 1 or greater.
 - MongoDB 4.0.y where y is 3 or greater.

If either of the controls fails, the broader test for the node will fail and the process will exit. The results of the test should be saved to `/tmp`.

```

Confirming upgrade qualifications

- [elastic-1] upgrade check complete: ✓
- [elastic-2] upgrade check complete: ✓
- [mongo-1] upgrade check complete: ✓
- [mongo-2] upgrade check complete: ✓

```

System Preparation and Bootstrap

Once `vbrick-setup` has the access it requires, it will confirm it is able to process your installation or upgrade and then start establishing the foundations necessary to complete that process.

```
Vbrick Setup will now copy all the packaging necessary to upgrade the
systems discovered during the configuration phase.

Checking free space on the remote devices

- [qaes01] 2GB min free partition space: ✓
- [qaes02] 2GB min free partition space: ✓
- [qamongo01] 2GB min free partition space: ✓
- [qamongo02] 2GB min free partition space: ✓

Updating package content on each node

- [qaes01] copy upgrade packaging: ✓
- [qaes02] copy upgrade packaging: ✓
- [qamongo01] copy upgrade packaging: ✓
- [qamongo02] copy upgrade packaging: ✓

Initializing environment on each node

- [qaes01] bootstrap completed: ✓
- [qaes02] bootstrap completed: ✓
- [qamongo01] bootstrap completed: ✓
- [qamongo02] bootstrap completed: ✓
```

This process can take several minutes per node because of the amount of content being copied. In addition, the bootstrap process will upgrade any prerequisites and do any system prep required to ensure a clear path for the next part of the process.

System Installation/Upgrade

Up until this step the process has been mostly the same; once we get to installation or upgrade however, the paths diverge. The processes are still very similar, in that a number of packages will get installed on each configured remote. The upgrade requires additional actions both pre and post upgrade and those differences are documented below.

Installation

`vbrick-setup` will pause after the System Preparation and Bootstrap is complete. Once you've hit enter to continue, the actual installation begins in earnest and will iterate through each service and node in that service until the process is complete.

```
Running installation on each node

- [vbrick-elastic-1] installed: ✓
- [vbrick-mongodb-1] installed: ✓

Running installation on this node

- [vbrick-haproxy] installing: ✓
```

The installation does take some time, but it significantly shorter than an upgrade.

Upgrade

If your systems have passed the **Upgrade Qualification Testing**, `vbrick-setup` will pause after [System Preparation and Bootstrap](#) and ensure that your Rev services have been stopped. Due to the nature of the upgrade, the Rev services must be fully stopped before you begin the installation of the Linux components. `vbrick-setup` will enforce this requirement by checking the listening ports on all the Rev servers listed in your configuration. If any of them are still online, `vbrick-setup` will pause and loop until this condition is changed.

A portion of this upgrade includes an upgrade to the structure of Elasticsearch indices and the output product is not compatible with the running version of Rev. In addition, there is a danger that Rev itself will try to alter these change so it is imperative that Rev remain stopped until you've completed the upgrade of not only the Linux components but the Windows components as well.

For information on stopping these services and preparing for the upgrade, see [Rev Upgrade Preparation](#).

```
Checking network status for Rev...

- qarev01 [10.150.2.16] Rev (port 80): ✓
- qarev02 [10.150.2.17] Rev (port 80): ✓

Hit enter to continue. █
```

As noted, the upgrade process will loop in this condition until it is certain Rev is not running before moving on to the next steps.

At this point, you will be given the option to backup both Elasticsearch and MongoDB.

```
Do you wish to backup Elasticsearch? (yes or no)
no

Do you wish to backup MongoDB? (yes or no)
no
```

Regardless of choice, the backups will be processed according to your selection. Additional details about backups is available in [Appendix C: Backup Elasticsearch and MongoDB](#). When the backup stage is passed, the upgrade process beings.

In this release the upgrade order is altered slightly so that each service is completely upgraded, server by server, until that service upgrade is complete (much like the manual instructions, although service-order is different). This may mean that in the case of arbiters, those servers will get processed twice; this is normal. The keeps each stage of the upgrade distinct and reduces the impact of a rollback or restore from backup should one be needed.

Elasticsearch is upgraded first and `vbrick-setup` will disable shard replication before the upgrade starts and then re-enable shard replication when the upgrade is complete. Depending on the architecture on your Elasticsearch cluster, it may take up to 10 minutes for Elasticsearch to clear its initial status. If the cluster is not yet green, `vbrick-setup` will wait for the status to update before proceeding.

```
Running upgrade on a per-service basis

Preparing Elasticsearch upgrade
- Disabling Shard replication: ✓

Upgrading each Elasticsearch node
- [qa-elastic-14a] upgraded: ✓
- [qa-elastic-14b] upgraded: ✓
- [qa-elastic-14c] upgraded: ✓

Elasticsearch post-upgrade processing
GREEN

- Enabling Shard replication: ✓
```

MongoDB is upgraded second. When the upgrade is complete `vbrick-setup` will add the `rev_audit` database user.

```
Preparing MongoDB upgrade

- Updating the MongoDB FeatureCompatibilityVersion flag to 4.2: ✓

Upgrading each MongoDB node
- [qa-mongo-14a] upgraded: ✓
- [qa-mongo-14b] upgraded: ✓
- [qa-mongo-14c] upgraded: ✓

MongoDB post-upgrade processing

Pausing for 180 seconds to allow MongoDB to settle.

- Updating the MongoDB FeatureCompatibilityVersion flag to 4.4: ✓
```

HAProxy upgrade is skipped in 7.40.

Note: For systems using `systemd` and `journald`

Systems that run `systemd` use `journald` as the replacement logging framework for `syslog`. These frameworks are standard on RHEL7 and Ubuntu 16.04 and may also be present in Ubuntu 14.04 but are not a default. Depending on your system, you may get warning or error messages from the HAProxy service when it starts because under almost all circumstances the Rev back-end services will be shut down, and these messages should reflect that state. Messages of following nature can be safely ignored until all Rev services are running:

```
Broadcast message from systemd-journald@qa-ha-15.lab.vb.loc (Fri 2018-11-16 20:05:40 UTC):
```

```
haproxy[5554]: backend rev_backend has no server available!
```

System Status

When the installation or upgrade is complete, the `vbrick-setup` will pause (120 seconds for installations, 240 seconds when OS upgrades took place) and then make sure the services are up and running. This is not a full test, just a network port check to make sure the services respond to remote requests.

```
Running service checks on each node
- [vbrick-elastic-1] testing port 9200: ✓
- [vbrick-mongodb-1] testing port 27017: ✓
- [vbrick-haproxy] testing this node, port 8081: ✓

The Vbrick Setup installation process for Linux services is complete.
```

If there are multiple nodes or arbiters in the configuration, these will be checked appropriately and reported.

```
Pausing for 240 seconds to allow the services to settle

Running service checks on each node
- [qaes01] testing port 9200: ✓
- [qaes02] testing port 9200: ✓
  MongoDB arbiter, testing port 27017: ✓
- [qamongo01] testing port 27017: ✓
- [qamongo02] testing port 27017: ✓
  Elasticsearch arbiter, testing port 9200: ✓
- [qaha] testing this node, port 8081: ✓

The Vbrick Setup upgrade process for Linux services is complete.

If the service tests were successful, you are safe to proceed with the
installation of your Rev node(s).
```

Next Steps - Post Installer

If you are installing Rev for the first time, proceed to [Rev Windows Components Installation](#).

If you just completed an upgrade, proceed to [Upgrade Rev Windows Components](#).

The `vbrick-setup` process makes every attempt to handle most issues so that any errors encountered will be truly *exceptional* and worth reporting back to engineering. Errors that happen up to, and during the bootstrap and configuration phase can usually be recovered from.

Errors that happen during the actual installation or upgrade will likely need manual intervention to move them forward. Every attempt has been made to wrap the process and cover most failures. If any errors were encountered during any part of this process, please report those errors immediately. Some basic troubleshooting and diagnostics can be found in [Post Installation Review](#).



Manual Installation and Upgrades - Getting Started

Last Updated: June 3, 2021

If you've chosen to handle your process manually, `vbrick-setup` can still be of some use. This section documents a number of considerations unique to a manual installation, most importantly for manual upgrades.

Manual versus Manual Assist

When it comes to a Linux system, a “manual installation” could mean anything. For purposes of this documentation “manual” is used to describe any installation or upgrade that is *not* fully automated and orchestrated across multiple devices (a feature offering of `vbrick-setup`). The `vbrick-setup` utility offers a “manual-assist” or “manual mode” feature that can improve on a completely bespoke installation. If something more customizable and hand-woven is desired, you can still use the self-extracting installer to put the package contents on the local disk of machines where it has been run and you can make use of those packages further.

A fully-manual or bespoke installation is outside of the scope of this documentation. Contact Vbrick Ops if you need further assistance. Documentation in this chapter is focused on manual, or “manual-assist” style processes. By default the self-extracting installer will leave the resources in `/var/lib/vbrick/7.40/files` and there is additional detail here on how to override that path. Packages for the supported distributions are in this location and should include most of the simple dependencies, and these can be installed by hand using the machine's package manager.

Note: The manual-assisted modes can be executed in any order however it is best to pick one service and iterate through all the nodes for that service before moving on to the next one.

For manual upgrades, it will help to process HAProxy first as it will provide the opportunity to upgrade your Elasticsearch indices.

Manual Installation Steps

If the destination nodes are customized or require individual attention, the manual process provides the ability to either install a new Rev service on a node-by-node basis. The manual process is the default of `vbrick-setup` and requires no arguments:

```
Manual installation
```

```
sudo vbrick-setup
```

Please see specific sections for Manual installation details:

- [Manual HAProxy Installation and Upgrade](#)

-
- [Manual Elasticsearch Installation and Upgrade](#)
 - [Manual MongoDB Installation and Upgrade](#)

Manual Upgrade Steps

If the destination nodes have been customized or require individual attention, the manual process provides the ability to upgrade existing nodes on a node-by-node basis.

Pre-Upgrade

Caution: Do not skip any of the pre-upgrade steps. Failure to perform any of the pre-upgrade steps may result in an inoperable system.

MongoDB: Feature Compatibility Version

Before upgrading MongoDB make sure you follow the steps detailed in [Manual MongoDB Installation and Upgrade](#) to [Retrieve and Update the Feature Compatibility Flag](#). Make certain the flag is set to **4.2**.

Note: All administrative mongo commands require authentication. Setting the **Feature Compatibility Version** is an administrative command.

Upgrade

The manual upgrader is meant to be useful in situations where manual upgrades have been run in the past. The process is unchanged in this release.

To perform a manual upgrade, you must add the `-u` or `--upgrade` flag:

```
Manual upgrade
sudo vbrick-setup -u
```

These upgrade steps are identical for each node and you repeat them on a service-by-service, node-by-node basis. Please see the specific per-service documentation for additional details:

- [Manual HAProxy Installation and Upgrade](#)
- [Manual Elasticsearch Installation and Upgrade](#)
- [Manual MongoDB Installation and Upgrade](#)

Post-Upgrade

MongoDB: Feature Compatibility Version

After upgrading MongoDB make sure you **repeat** the steps detailed in [Manual Installation and Upgrades - Getting Started](#) to [Retrieve and Update the Feature Compatibility Flag](#). Make certain the flag is set to **4.4**.

MongoDB: Add Rev Audit database user

This step is optional and required only ahead of when the feature is enabled however the upgrade window is a very good time to apply this change regardless of the immediate need. The specific instructions are outlined in [Appendix F: MongoDB Supplemental Information](#) -

Rev Audit Database User. This step may also be taken pre-upgrade, there is no strict need for the timing of this event outside of Rev's need for the user to exist when the feature is enabled.

Next Steps

Once you have successfully upgraded all your Linux servers in the installation/upgrade sections above, you can proceed with one of the following options:

- For installations, continue to the [Rev Windows Components Installation](#) document.
- For upgrades, continue to the [Upgrade Rev Windows Components](#) document.



Manual HAProxy Installation and Upgrade

Last Updated: June 10, 2021

This section describes the manual installation or upgrade of HAProxy service. If you used the **Automatic Setup** option there is no reason to use the processes documented here.

In some situations, working through a configuration node by node is preferable, and manual installation gives you that option. The `vbrick-setup` process still requires a functional `vbrick-vars.txt` and will chose the appropriate service based on the IP address of the node you are running the process on and its role in your configuration.

Caution: Please make sure you check the [SSL](#) section later in this topic to familiarize yourself with the cipher and SSL settings Vbrick applies by default during an installation..

What to Expect

Manual, or per-node installations do not have to establish any remote connectivity or move content onto each node in turn so this process is very quick. If the prerequisites are met, and the machine you are running the process on matches something in your configuration, `vbrick-setup` will perform the appropriate installation for that role. In the past there were menu options for selecting this, but given that the end user will (or should) have edited a configuration file before this step, the role is already implied and a menu choice should be unnecessary.

If `vbrick-setup` cannot match an IP address on the running machine to one in the configuration, the process will exit.

Using `vbrick-setup`

In this release you can now perform either a manual installation or a manual upgrade, depending on your needs. The `vbrick-setup` program should have already been installed as documented in [Installer File Downloads and Configuration](#). If you've skipped by this step, return now and complete the process by using the self-extracting installer.

Manual Installation

Manual installation is part of the same package and process that includes the automatic setup features. The `vbrick-setup` program has already been installed as documented in [Installer File Downloads and Configuration](#). If you skipped by this step, return now and complete the process by using the self-extracting installer.

The installer is run with the following command:

```
Run the installer
sudo vbrick-setup
```

The installer will make every attempt to find your copy of `vbrick-vars.txt`, parse that into a working set of orchestration details, and then present them to you so that you can see your configuration as it is intended to be delivered. When you confirm that the node you're on will install the services you intend, hit the **Enter** key and the process will run until completion.

```
Performing installation for HAProxy: ✓
```

Manual Upgrade

On-Premises Rev v7.40 does not contain an upgrade for HAProxy.

Verifying an HAProxy Installation

By default Vbrick adds a status service to the HAProxy installation - this service is a series of configuration settings that are part of HAProxy, and we simply make sure those settings are enabled. The statistics service can be accessed from a web browser using the following URL:

`https://<IP Address of the HAProxy node>:8081/`

The SSL certificate that is installed is self-signed certificate for `*.example.com` which *will* cause browser warning messages regarding the quality of the certificate. For purposes of this interface, the self-signed SSL is sufficient and will protect the username and password fields used to login. When you browse to this page you'll need the following credentials:

- **Username:** vbrick
- **Password:** VBrickHAProxy1

The HAProxy status page lists all the connection points in the configuration. Each listening port has a front-end for the listener and a backend for the service distribution. The names in the tables should match the names you used in `vbrick-vars.txt` and should have a green status. Because this is the first node installed, and the Rev Runtime nodes are the last, you can expect this status to remain red until the entire installation process is complete and the Rev Runtime service is running.

Once the Rev Runtime service is green in the HAProxy status page, you can further test the HAProxy service by using the HTTP/80 service by DNS name to initialize and start using Rev.

SSL

If you intend to deliver your application via HTTPS, you will want to replace the SSL certificate installed in `/etc/haproxy/cert.pem` - the PEM format is a combined Private Key and Public Certificate.

SSL certificate generation is documented in [Appendix E: SSL Certificates](#).

Vbrick has altered the SSL configuration it distributes with HAProxy.

- The string `:!3DES:!RC4` is appended to the cipher string removing support for 3DES and RC4 ciphers.
- The strings `no-tlsv10` and `no-tlsv11` have been added to the SSL-specific configuration string, removing support for **TLS v1.0** and **v1.1**.

Next Steps

If you've completed the HAProxy installation, continue to the [Manual Elasticsearch Installation and Upgrade](#) section.

HAProxy Installation Service Specifications

Files

File	Path
Configuration	/etc/haproxy/haproxy.cfg
SSL Certificate	/etc/haproxy/cert.pem
Logging	The configuration provided by the installer configures logging to use the syslog channel LOCAL_0 which under most circumstances will be delivered to the main system log unless a specific allowance is made in the syslog configuration. The HAProxy configuration can also be altered to use an alternate location for logs

Network

Service Port	Description
80	The HTTP port for the Rev Runtime.
443	The HTTPS port for the Rev Runtime. The SSL for this port is governed directly by the HAProxy configuration and as delivered uses a self-signed certificate.
8081	An HTTPS status service port, used to monitor the state of HAProxy.



Manual Elasticsearch Installation and Upgrade

Last Updated: June 3, 2021

This document describes the manual installation or upgrade of the Elasticsearch service. If you used the **Automatic Installation and Upgrades** option there is no reason to use the processes documented here.

In some situations, working through a configuration node by node is preferable, and manual installation or upgrade gives you that option. The `vbrick-setup` process requires a local, functional `vbrick-vars.txt` and will chose the appropriate service based on the IP address of the node you are running the process on and its role in your configuration.

Warning: If you've chosen a manual installation then the Vbrick Rev nodes *must* be installed one at a time and in sequence as they appear in the manual installation instructions. If you have not already installed HAProxy, please return to the [Manual HAProxy Installation and Upgrade](#) section and complete those steps before returning to the process documented below.

What to Expect

Manual, or per-node installations don't have to establish any remote connectivity or move content onto each node in turn so this process is very quick. If the prerequisites are met, and the machine you are running the process on matches something in your configuration, `vbrick-setup` will perform the appropriate installation for that role. In the past there were menu options for selecting this, but given that the end user will or should have edited a configuration file before this step, the role is already implied and a menu choice should be unnecessary.

Arbiters

If you intend to have only two primary Elasticsearch nodes, a third Elasticsearch arbiter will be installed one of the MongoDB servers (usually the second) so you do not need to worry about that process in this step. Elasticsearch arbiters are detected by the configuration in `vbrick-vars.txt` - more details about Arbiters are available in this document: [Linux Cluster Considerations](#).

Additionally, an Elasticsearch node may also serve as a MongoDB node and this configuration will be detected and managed automatically. There are no additional steps beyond providing a configuration that describes this type of architecture.

Using `vbrick-setup` During an Elasticsearch Installation

The `vbrick-setup` process allows you to perform either a manual installation or manual upgrade depending on your needs. The `vbrick-setup` program should have already been installed as documented in [Installer File Downloads and Configuration](#). If you skipped this step, return now and complete the process by using the self-extracting installer.

Elasticsearch Manual Installation

Manual installation is part of the same package and process that includes the automatic setup features.

The installer is run with the following command:

```
Run the installer
```

```
sudo vbrick-setup
```

The installer will make every attempt to find your copy of `vbrick-vars.txt`, parse that into a working set of orchestration details, and then present them to you so that you can see your configuration as it is intended to be delivered. When you confirm that the node you're on will install the services you intend, hit the **Enter** key and the process will run until completion.

Make sure to verify your installation. See: [Verifying an Elasticsearch Installation](#).

Elasticsearch Manual Upgrade

The manual upgrade process has near identical task structure to the installation task, with the addition of upgrade qualification testing. Upgrades must be toggled intentionally, `vbrick-setup` will not try to detect the status when it is not running in automatic mode. To toggle an upgrade, use the `-u` or `--upgrade` flag when running `vbrick-setup`:

```
Run the installer with the upgrade flag
```

```
sudo vbrick-setup -u
```

Make sure to verify your installation. See: [Verifying an Elasticsearch Installation](#)

Verifying an Elasticsearch Installation

There are multiple ways to check the health of the Elasticsearch service. The output of queries to Elasticsearch are in JSON format. If you've just brought up a node or an entire cluster, keep in mind the following:

- The Vbrick Elasticsearch configuration defines both a minimum recovery time, and a required minimum number of cluster nodes to be online before cluster operations can continue. As a result, Elasticsearch cluster status may remain `red` for at least ten minutes after installation has completed. This is especially true in clusters so please allow at least this amount of time to pass before you move on to the [Elasticsearch Troubleshooting](#) section.

Command Line Status

Log into any Elasticsearch node that has the service running. Use the command below to poll local status:

```
Poll local status
```

```
curl -XGET 'http://localhost:9200/_cluster/health?pretty=true'
```

The output below reflects a healthy 2 Data Node, 1 arbiter Elasticsearch cluster:

```
{
  "cluster_name" : "rev",
```



```

    "status" : "green",
    "timed_out" : false,
    "number_of_nodes" : 3,
    "number_of_data_nodes" : 2,
    "active_primary_shards" : 49,
    "active_shards" : 98,
    "relocating_shards" : 0,
    "initializing_shards" : 0,
    "unassigned_shards" : 0,
    "delayed_unassigned_shards" : 0,
    "number_of_pending_tasks" : 0,
    "number_of_in_flight_fetch" : 0,
    "task_max_waiting_in_queue_millis" : 0,
    "active_shards_percent_as_number" : 100.0
  }

```

Web Browser Status

Open a browser and access the following URL:

`http://<ELASTICSEARCH_DNS_NAME_OR_IP>:9200/_cluster/health?pretty=true`

The output in the web browser will present the same data as the command-line example above.

Verify the Results

Using either the command line or the web browser will yield identical data. Use the table below to compare your results.

Attribute	Value
cluster_name	Should be rev
status	One of red , yellow , or green
timed_out	Should be false
number_of_nodes	Determined by configuration; 3 for multi-node and 1 for single-node
number_of_data_nodes	Determined by configuration; multi-node clusters without arbiters should be 3 ; multi-node clusters with arbiters should be 2 while single node systems should be 1 .

Note: Only the **red** status can be considered one for major concern because processes that rely on the service will not be able to communicate with the service in a useful way (network ports are open but errors are returned). Once an individual instance or cluster clears the configured minimums for nodes and recovery time, the status should return to at least **yellow**. Processes that rely on the service will be able to make index queries with a status of **yellow** or **green**.

Next Steps

Repeat this process for each primary Elasticsearch node (do not attempt to force an arbiter installation). If you've completed all your primary Elasticsearch installations, continue to the [Manual MongoDB Installation and Upgrade](#) section.

Elasticsearch Installation Service Specifications

Files

File	Path
Configuration	/etc/elasticsearch/elasticsearch.yml
Logging	/var/log/elasticsearch/rev.log
Data	/var/lib/elasticsearch

Network

Service Port	Description
9200	Main service port, must be open between all Elasticsearch nodes and the Rev nodes.
9300	Backend communication port, must be open to all Elasticsearch nodes.

Manual MongoDB Installation and Upgrade

Last Updated: June 3, 2021

This document describes the manual installation of the MongoDB service. If you used the Automated Installation and Upgrades option there is no reason to use the processes documented here.

In some situations, working through a configuration node by node is preferable, and manual installation gives you that option.

Caution: vbrick-vars.txt

The manual `vbrick-setup` process requires a local, functional `vbrick-vars.txt` and will choose the appropriate service based on the IP address of the node you are running the process on and its role in your configuration.

Note: MongoDB authentication was turned on in 7.25. If you need to interact with MongoDB on the command line, make sure you are using suitable authentication parameters to gain authenticated access.

More details about this process can be found here: [Appendix F: MongoDB Supplemental Information](#).

Caution: If you have chosen a manual installation then the Vbrick Rev nodes *must* be installed one at a time and in sequence as they appear in the manual installation instructions.

If you have not already installed Elasticsearch, please return to the [Manual Elasticsearch Installation and Upgrade](#) documentation and complete those steps before returning to the process documented below.

What to Expect

Manual, or per-node installations, do not have to establish any remote connectivity or move content onto each node in turn so this process is very quick. If the prerequisites are met, and the machine you are running the process on matches something in your configuration, `vbrick-setup` will perform the appropriate installation for that role. In the past there were menu options for selecting this, but given that the end user will (or should have) edited a configuration file before this step, the role is already implied and a menu choice should be unnecessary.

Arbiters

If you intend to have only two primary MongoDB nodes, a third MongoDB arbiter will be installed one of the Elasticsearch servers (usually the second) so you do not need to worry about that process in this step. MongoDB arbiters are detected by the configuration in `vbrick-vars.txt` - more details about Arbiters are available in this document: [Linux Cluster Considerations](#).

Additionally, a MongoDB node may also serve as an Elasticsearch node and this configuration will be detected and managed automatically. There are no additional steps beyond providing a configuration that describes this type of architecture.

Using vbrick-setup During MongoDB Installation

The `vbrick-setup` process allows you to perform either a manual installation or manual upgrade depending on your needs. The `vbrick-setup` program should have already been installed as documented in [Installer File Downloads and Configuration](#). If you have skipped this step, return now and complete the process by using the self-extracting installer.

MongoDB Manual Installation

The manual installation is part of the same package and process that includes the automatic setup features.

The installer is run with the following command:

```
Run the installer
```

```
sudo vbrick-setup
```

The installer will make every attempt to find your copy of `vbrick-vars.txt`, parse that into a working set of orchestration details, and then present them to you so that you can see your configuration as it is intended to be delivered. When you confirm that the node you're on will install the services you intend, hit the **Enter** key and the process will run until completion.

Make sure to verify your installation. See: [Verifying a MongoDB Installation](#).

MongoDB Manual Upgrade

Important! Make sure you review the [Retrieve and Update the Feature Compatibility Flag](#) section before you proceed with a manual upgrade!

The manual upgrade process has near identical task structure to the installation task, with the addition of upgrade qualification testing. Upgrades must be toggled intentionally, `vbrick-setup` will not try to detect the status when it is not running in automatic mode. To toggle an upgrade, use the `-u` or `--upgrade` flag when running `vbrick-setup`:

```
Run the installer with the upgrade flag
```

```
sudo vbrick-setup -u
```

Make sure to verify your installation. See: [Verifying a MongoDB Installation](#).

Retrieve and Update the Feature Compatibility Flag

Each version of MongoDB has an associated versioning flag for the WiredTiger storage engine that enables the features of that version. In the 7.40 release, it is necessary to check and set this flag in *advance* of the upgrade.

To update the Feature Compatibility Version flag, either locate the current master MongoDB server or, using the single instance, log into that machine. Once you have access to the **primary MongoDB server**, use the mongo shell to retrieve the current feature compatibility version (we expect it to be 4.0):

Step 1: Retrieve current feature compatibility version flag setting - expected 4.0

```
rev:PRIMARY> db.adminCommand( { getParameter: 1,
featureCompatibilityVersion: 1 } )
{
  "featureCompatibilityVersion" : {
    "version" : "4.0"
  },
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1619635793, 1),
    "signature" : {
      "hash" : BinData(0,"hIhsypbq80vkMSeAxmjd narFS3k="),
      "keyId" : NumberLong("6954359158183297026")
    }
  },
  "operationTime" : Timestamp(1619635793, 1)
}
```

If the value returned is **4.0** as expected (and seen above), then the next step is to update it to **4.2**.

Step 2: Update current feature compatibility version flag- set to 4.2

```
rev:PRIMARY> db.adminCommand( { setFeatureCompatibilityVersion: "4.2" } )
{
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1619635895, 1),
    "signature" : {
      "hash" : BinData(0,"/Es/cqbzb9urrrmIo/wDr6W4a9c="),
      "keyId" : NumberLong("6954359158183297026")
    }
  },
  "operationTime" : Timestamp(1619635895, 1)
}
```

Now run the manual upgrade process on *each* MongoDB node.

Once complete, relocate the master node and then execute the following command from within the Mongo shell:

Post Upgrade Step: Update Master node feature compatibility version flag- set to **4.4**

```
rev:PRIMARY> db.adminCommand( { setFeatureCompatibilityVersion: "4.4" } )
{
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1622663603, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  },
  "operationTime" : Timestamp(1622663603, 1)
}
```

Note: You can use the `db.adminCommand({ getParameter: 1, featureCompatibilityVersion: 1 })` command in the mongo shell to confirm that you have set this value correctly at any time.

Verifying a MongoDB Installation

The Vbrick installation for MongoDB installs a command line tool, appropriately named `mongo` that can communicate with a running MongoDB server. Bear in mind that authentication for MongoDB is turned on so you need to have your user name and password handy.

For a single-node instance, the simplest test for MongoDB is to connect to the device, and from an unprivileged shell type:

MongoDB verification

```
sudo mongo --username admin --password --authenticationDatabase admin
```

In the example above, we're using the username **admin** and allowing the password to be entered interactively. The output of commands in the MongoDB client is JSON formatted, and some responses can be long, depending on the query.

You can confirm the version (4.0.3) of the MongoDB server with the following command from within the MongoDB CLI:

Confirm MongoDB version

```
db.version()
```

For single-node users, provided you enter your password correctly for the `sudo`, the `mongo` command will either succeed or fail to connect to the database, and that is sufficient health status for the purposes of the On-Premises installers.

Note: Use `ctrl + d` at any time to exit the MongoDB client.

If you install a group of MongoDB servers, this command is the best way for establishing a connection with a running MongoDB instance. To further validate your cluster, first establish

your connection with either of the **primary** MongoDB nodes (you can use the arbiter for health checks, but it is a better practice to work with either the **primary** or **secondary** servers in the cluster).

When you use the MongoDB client on replicated instances, the prompt will reflect that node's status in the cluster. On nodes with healthy replication, the prompts should be one of the following:

- rev:PRIMARY>
- rev:SECONDARY>
- rev:ARBITER>

Once connected, use the commands below to query the replication status:

Query replication status

```
rs.status()
```

Below is an example of what you could expect to see:

```
{
  "set" : "rev",
  "date" : ISODate("2018-04-12T18:09:13.415Z"),
  "myState" : 1,
  "term" : NumberLong(7),
  "heartbeatIntervalMillis" : NumberLong(2000),
  "optimes" : {
    "lastCommittedOpTime" : {
      "ts" : Timestamp(1523556552, 5),
      "t" : NumberLong(7)
    },
    "appliedOpTime" : {
      "ts" : Timestamp(1523556552, 5),
      "t" : NumberLong(7)
    },
    "durableOpTime" : {
      "ts" : Timestamp(1523556552, 5),
      "t" : NumberLong(7)
    }
  },
  "members" : [
    {
      "_id" : 0,
      "name" : "MONGO-01:27017",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 3802988,
      "optime" : {
        "ts" : Timestamp(1523556552, 5),
        "t" : NumberLong(7)
      },
      "optimeDate" : ISODate("2018-04-12T18:09:12Z"),
```

```

        "electionTime" : Timestamp(1519753610, 1),
        "electionDate" : ISODate("2018-02-27T17:46:50Z"),
        "configVersion" : 48421,
        "self" : true
    },
    {
        "_id" : 1,
        "name" : "MONGO-02:27017",
        "health" : 1,
        "state" : 2,
        "stateStr" : "SECONDARY",
        "uptime" : 3802924,
        "optime" : {
            "ts" : Timestamp(1523556538, 12),
            "t" : NumberLong(7)
        },
        "optimeDurable" : {
            "ts" : Timestamp(1523556538, 12),
            "t" : NumberLong(7)
        },
        "optimeDate" : ISODate("2018-04-12T18:08:58Z"),
        "optimeDurableDate" : ISODate("2018-04-12T18:08:58Z"),
        "lastHeartbeat" : ISODate("2018-04-12T18:09:11.568Z"),
        "lastHeartbeatRecv" : ISODate("2018-04-
12T18:09:13.385Z"),
        "pingMs" : NumberLong(0),
        "syncingTo" : "QAMONGO01:27017",
        "configVersion" : 48421
    },
    {
        "_id" : 2,
        "name" : "MONGO-03:27017",
        "health" : 1,
        "state" : 7,
        "stateStr" : "ARBITER",
        "uptime" : 3802953,
        "lastHeartbeat" : ISODate("2018-04-12T18:09:11.568Z"),
        "lastHeartbeatRecv" : ISODate("2018-04-
12T18:09:12.836Z"),
        "pingMs" : NumberLong(0),
        "configVersion" : 48421
    }
},
"ok" : 1
}

```

In the example above, there are 2 data nodes (MONGO-01, and MONGO-02), and one arbiter (MONGO-03).

- In the root of the JSON tree there is a **members** is a JSON array that consists of the number of members configured in this replica set. You should see three members.

- The **stateStr** attribute value of each node should read one of **PRIMARY**, **SECONDARY**, or **ARBITER**
- The **health** attribute value of each node should read 1
- The **lastHeartbeat** attribute value of each node should be a recent timestamp.
- The last second-to-last line of the output should read **“ok” : 1**.

Note: A MongoDB client prompt or stateStr of **STARTUP** or **STARTUP2** is a sign of a transitional state and should resolve within a few minutes.

Caution: A MongoDB node that fails to transition to one of **PRIMARY**, **SECONDARY**, or **ARBITER** after several minutes will likely require manual intervention. Although more detail about a possible cause will be in the JSON output of `rs.status()`, diagnosing MongoDB replication problems is outside of the range of this documentation.

For more information please refer to the [MongoDB Replica Set Member States](#) documentation.

For additional troubleshooting of this installation, see: [Troubleshooting the Installation](#).

Next Steps

Repeat this process for *each* primary MongoDB node (do not attempt to force an arbiter installation).

If you've completed *all* your MongoDB tasks, proceed with one of the following options:

- For new installations, continue to the [Rev Windows Components Installation](#) topic in the **Automated Installation** section.
- For upgrades, continue to the [Upgrade Rev Windows Components](#) topic in the **Automated Installation** section.

MongoDB Installation Service Specifications

Files

File	Path
Configuration	/etc/mongod.conf
Logging	/var/log/mongod/mongod.log
Data	Red Hat: /var/lib/mongo Ubuntu: /var/lib/mongodb

Network

Service Port	Description
27017	Primary communications port for server process mongod.



Rev Windows Components Installation

Last Updated: June 3, 2021

This section describes the installation of the Vbrick Rev Runtime Service.

Caution: Rev Runtime should *not* be installed with an employee's account. This ensures that if that particular employee leaves, some components of Rev will not need to be reinstalled as a result. Please make sure you review the requirements for [Windows Service Accounts](#).

The Vbrick Rev nodes are designed to be installed *one at a time* and *in sequence* as they appear in these installation instructions; Please do not attempt to install more than one node at a time.

Note: Rev Runtime can support both local and remote file systems for the storage of content. This document uses UNC paths, for example,

```
\\localfileserver\revstorage
```

However, Rev can support other types of remote storage as well. Please see [Appendix B: File Storage Path Configuration](#) for more information.

Dependencies

It is important that you review the [Installation Prerequisites](#) topic which covers machine sizing, setting up DNS, and other important information before beginning or completing the installation. If your Windows server does not have .Net framework 4.7.2 or greater, please install or enable .Net framework 4.7.2 runtime. More information on .Net Framework can be found at <https://dotnet.microsoft.com/download/dotnet-framework/net472>.

Before You Begin

You will need to note the following attributes about your environment:

- **Vbrick Rev VIP Address** - The address your load balancer presents as your Vbrick Rev Cluster
- **Vbrick Rev Node 1 Address** - IP Address and Hostname in UPPERCASE LETTERS (without domain). *This must match the Computer name in the Windows System Control Panel.*
- **Vbrick Rev Node 2 Address** - IP Address and Hostname in UPPERCASE LETTERS (without domain). *This must match the Computer name in the Windows System Control Panel.*
- **Elasticsearch Node 1 Address** - IP Address and Hostname (without domain)
- **Elasticsearch Node 2 Address** - IP Address and Hostname (without domain)
- **MongoDB Node 1 Address** - IP Address and Hostname (without domain)
- **MongoDB Node 2 Address** - IP Address and Hostname (without domain)

-
- **MongoDB Username and Password** - A suitable default is provided by the Vbrick Rev Installer
 - **Shared Storage** - A remote volume that is reachable from Vbrick Rev Nodes 1 and 2 for the storage of videos. There is additional documentation about shared storage in [Appendix B: File Storage Path Configuration](#).
 - **Erlang Cookie** - A suitable default is provided by the Vbrick Rev Installer

All of the above requirements *must* be met before you may proceed with this section of the installation.

Note: It is recommended that if the Microsoft Windows nodes have had FIPS encryption enabled it should be disabled before you begin installing the Vbrick Rev Service.

Windows Service Accounts

Rev Runtime and its Windows Components require administrative-level access to the systems they run on. In addition, any remote filesystem in use by Rev Runtime *must* also have **read** and **write** permissions for the same service-user or the service may fail to operate. The service account used for Rev Runtime should not be tied directly to an employee or otherwise transient account that might have a different lifespan from Rev Runtime. In simple Windows Server deployments, you can either directly use the main Administrator account or create and then enable that account as an administrative account.

In Windows Server environments that leverage Active Directory or Windows NT Domains the requirements are still similar. Active Directory security has too many permutations to document here however the critical requirements are simple: a single system user that belongs to a group (or multiple groups as needed) capable of managing services local to the devices they are run on. To be clear, full network administrative rights are not a requirement, only service and system rights local to those systems where Rev Runtime will run. This same user will also require the read and write permissions for any shared storage used by these environments.

Make sure you use this Service Account to access these systems and run the installer in the next step.

Install Vbrick Rev Software

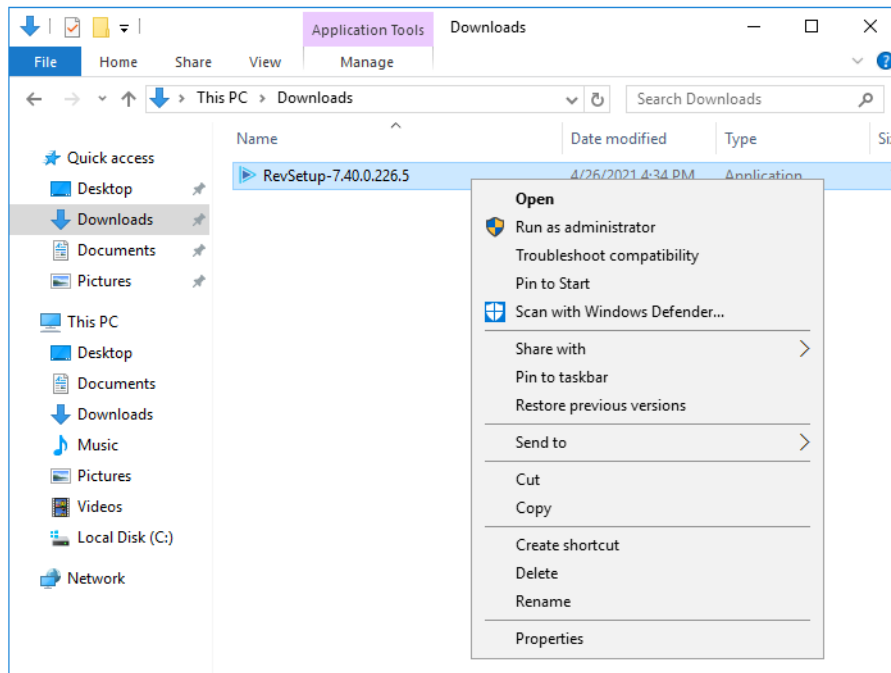
The Vbrick Rev Service is installed on two Microsoft Windows Server nodes. In this documentation these two nodes are named **REVNODE1** and **REVNODE2** and **must** be in UPPERCASE LETTERS.

Perform the installation on the **Primary Node** first.

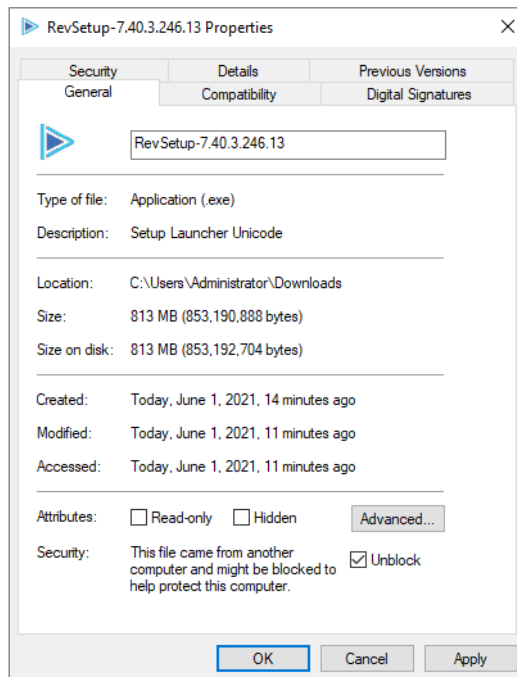
Caution: The Primary Vbrick Rev node should be installed **first**. The Secondary Vbrick Rev node should be installed after this entire process is completed on the Primary node.

1. Log into the Vbrick Rev Microsoft Windows Server node as a Local Administrator. If a Domain Account is to be used, please contact Vbrick Support for assistance.
2. Open the Windows System Control Panel and note the Computer name. This name *must* match the **Rev Node Name** used later in the installation process.
3. Copy the `Vbrick_RevSetup-7.40.X.Y.exe` file to this computer.

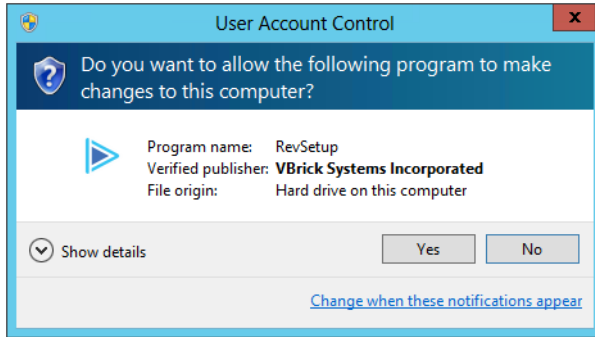
- Right-click on the file and click **Properties**.



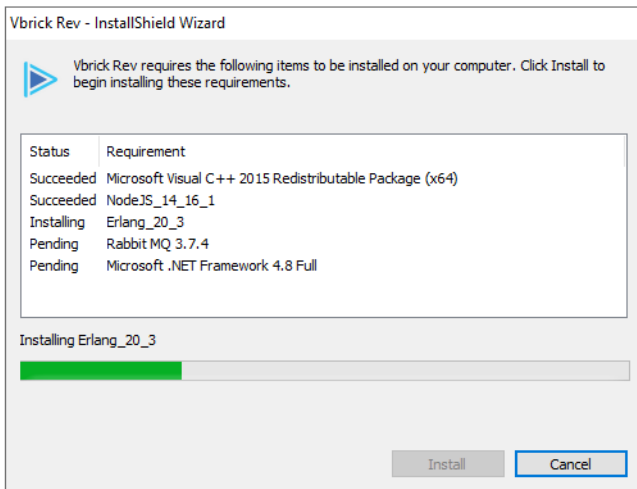
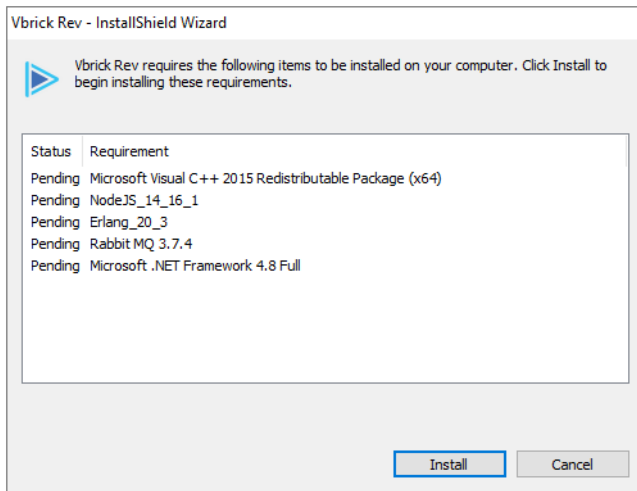
- If there is a button named **Unblock**, click it and then click **OK**.



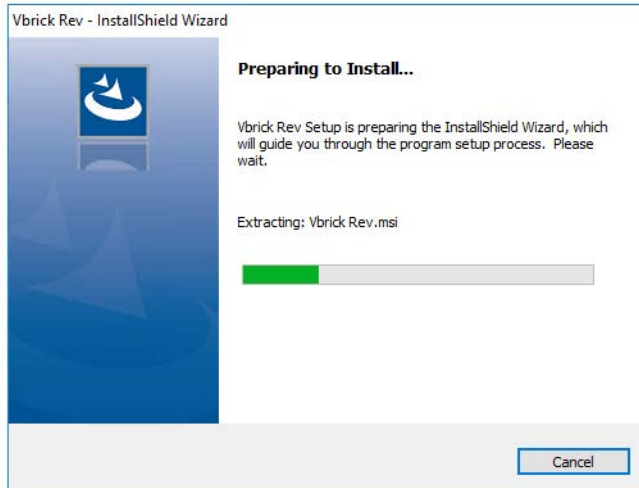
- Right-click on the file again and select **Run as administrator**.
- If prompted, at the **User Account Control** window, click the **Yes** button to allow the program to run.



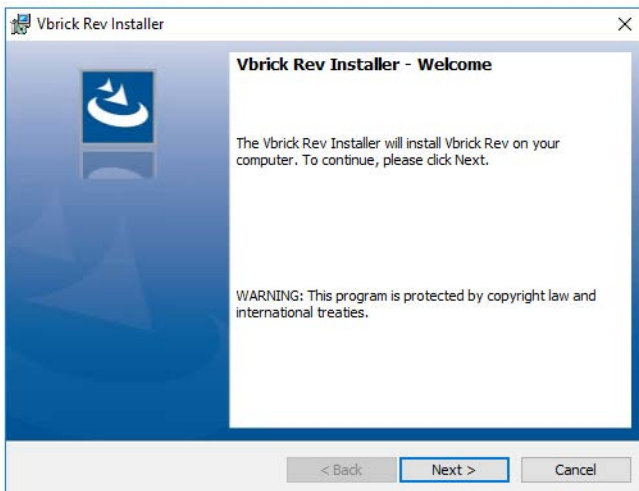
8. The Vbrick Rev Setup program will first install some required components. Select **Install** to continue.



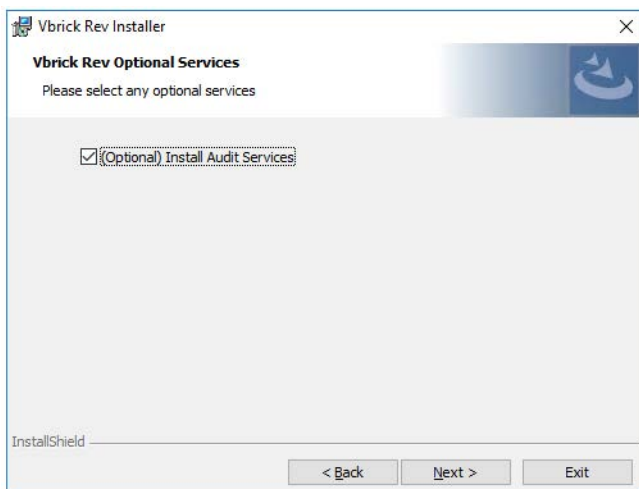
9. Restart when prompted. After restarting, the Vbrick Rev Setup program will automatically continue with the installation.



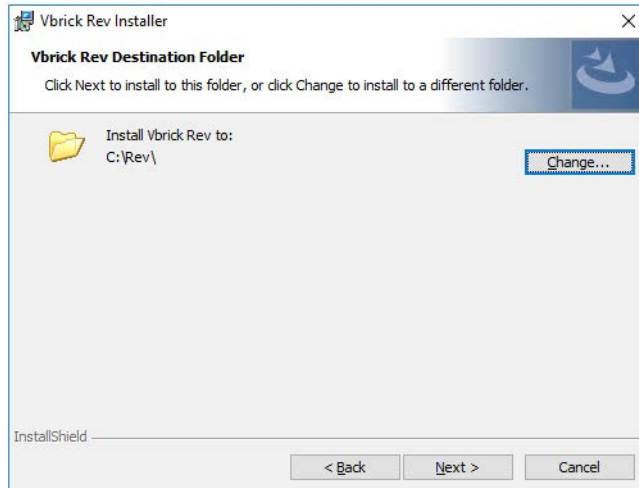
10. Click **Next** in the Vbrick Rev Installer window to continue.



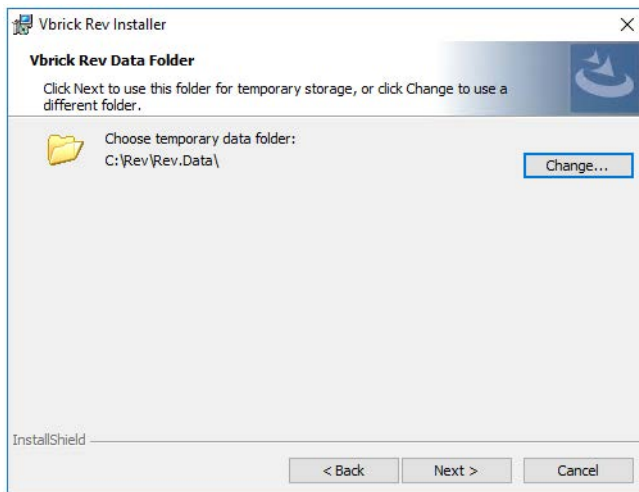
11. Select **Install Audit Services** if you want to install the Audit Service. Otherwise, click **Next** to continue. See [Rev Audit Services Configuration - Installation](#) for more details.



12. Choose the **Vbrick Rev Destination Folder** which contains the Vbrick Rev program files. Click **Change** to select another location.



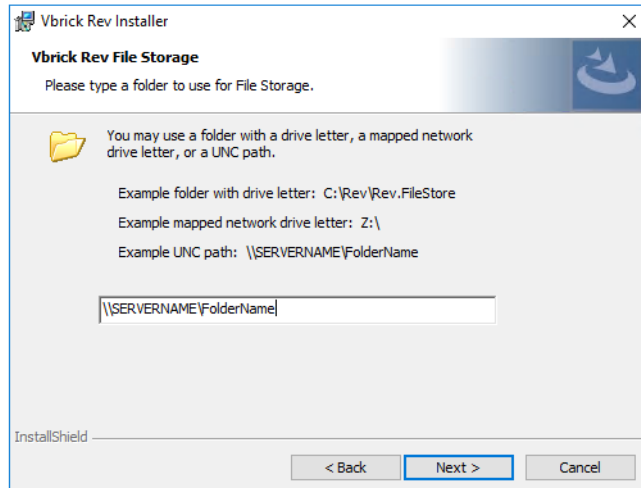
13. Choose the **Vbrick Rev Data Folder** which contains temporary data that Vbrick Rev uses. Click **Change** to select another location.



14. Choose the **Vbrick Rev File Storage Folder** which Vbrick Rev uses to store video files. You may use a folder with a drive letter, a mapped network drive letter, or a Windows UNC path.

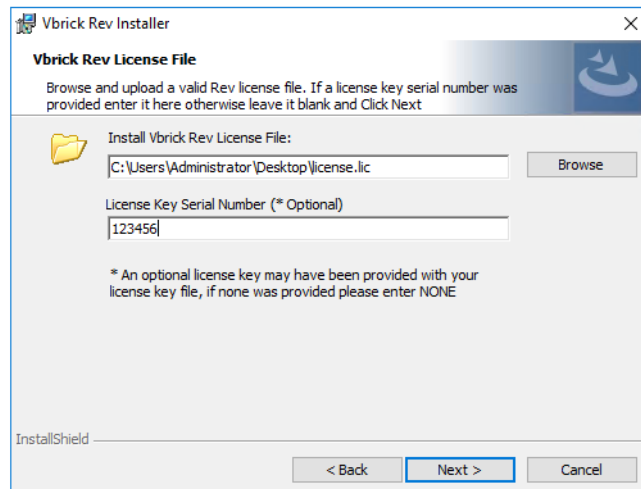
Caution: Remote File Storage Paths - This document uses Windows UNC paths in the examples however there are other methods for configuring remote storage. Make sure you've reviewed these options in [Appendix B: File Storage Path Configuration](#) before you continue with the configuration. Depending on your network filesystem, the values you will use in the setup will necessarily be different from those described here so it is critical that you apply the appropriate configuration at this time.

Note: Any file system location that you specify for storage *must* exist before you attempt to start Vbrick Rev Service. The Vbrick Rev Installer will not create the folder if it does not already exist.



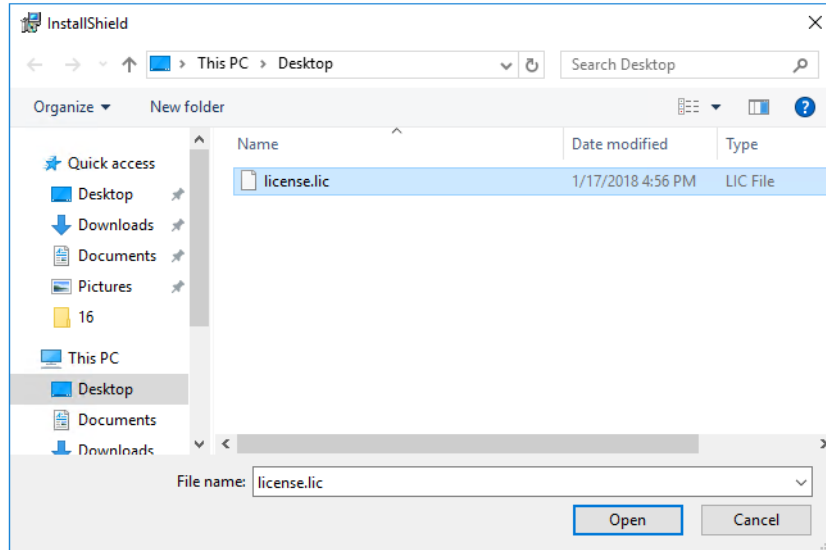
Caution: Windows Service Account Access: The account used to access the server running this installer *must* have read and write permissions. Additional details are available in [Windows Service Accounts](#).

15. Click **Browse** to choose the location of the Vbrick Rev License File. If you don't have one, click **Next**.

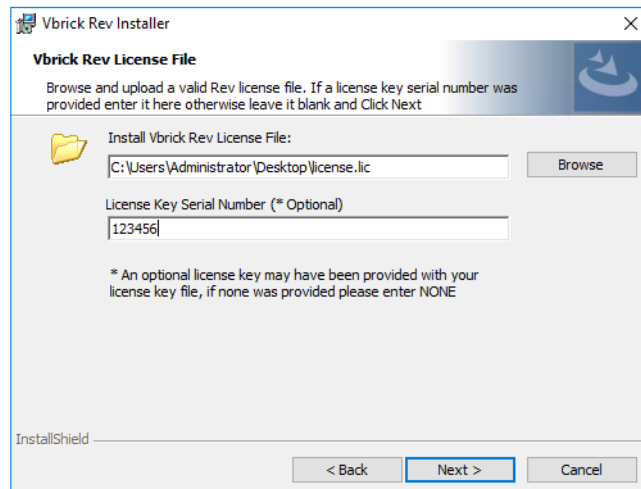


Caution: If you do not enter a License File you will not be able to start the Vbrick Rev License Service nor create the root account and activate Rev.

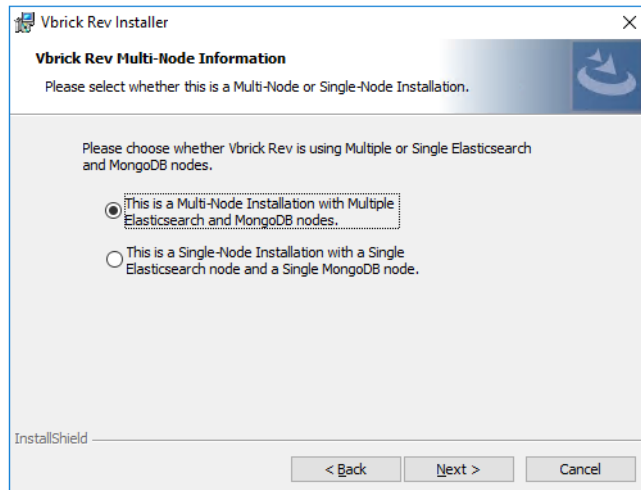
16. Select the **License File** provided by Vbrick and click **Open**.



17. If you have been provided a **License Key Serial Number** with your license key, enter it in **License Key Serial Number** field. This serial number should match exactly as it appears from the file provided by Vbrick Support. Not all license keys require a serial number depending on the type of license key you have been provided.

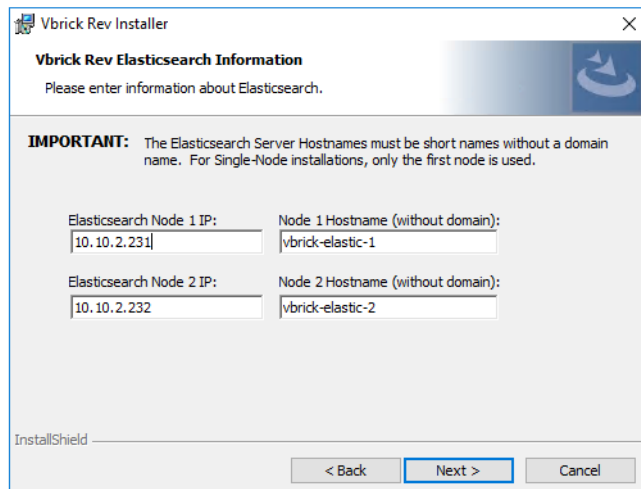


18. Decide whether this will be a **Multi-Node** or a **Single-Node** installation. In a Single-Node installation, only the first value for Elasticsearch and MongoDB are relevant.



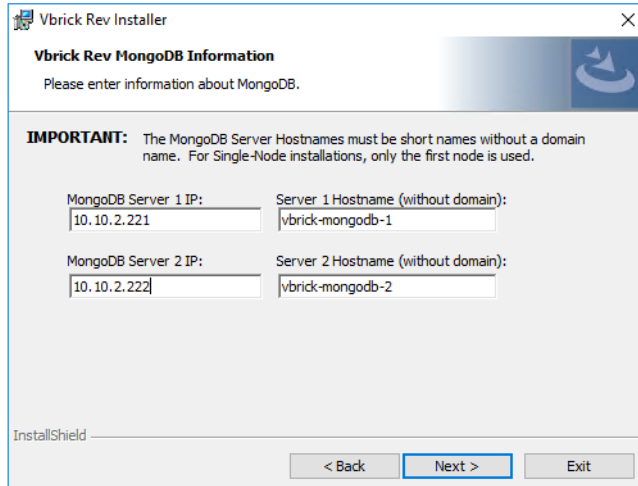
19. Enter the **Elasticsearch Cluster** information.

- The Elasticsearch Node IP and Hostname fields should be changed to reflect the values in your environment.



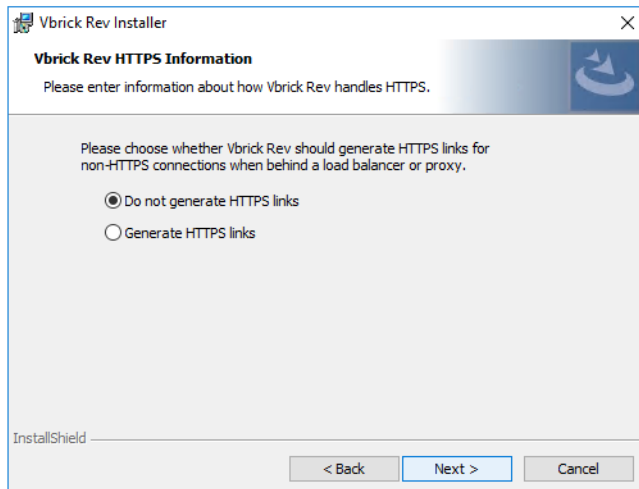
20. Enter the **MongoDB Cluster** information.

- The MongoDB Server IP and Hostname fields should be changed to reflect the values in your environment.
- The MongoDB Server Host names must consist of the host name *without* the domain name.



21. Choose how Vbrick Rev should be configured to serve HTTPS (secure) links.

- While HTTPS is not a direct requirement Vbrick strongly suggests it be used whenever possible. The Rev interface has a number of places where credentials are required and when these are open to clear-text (non-encrypted) data entry they will fail the most rudimentary security scans. Simply stated, SSL and HTTPS are best practices in any situation where sensitive data might be exchanged in a web browser over a network.
- This step determines the URL scheme Rev Runtime will use to generate links to content. HTTPS is secured HTTP and noted by URLs that start with https://. The content in pages with this URL are encrypted with TLS/SSL. In simple and clustered installations, SSL encryption must be offloaded to a load balancer such as HAProxy, which acts as the HTTPS endpoint. The Vbrick Installer's default is for Vbrick Rev to not generate HTTPS links.



Note: If HTTPS is required, configure the load balancer as the HTTPS endpoint (port 443) with the load balancer connecting to Vbrick Rev using regular HTTP (port 80). You must choose Generate HTTPS links in this situation.

22. Enter **Vbrick Rev Cluster** information.

Vbrick Rev Cluster Information
Please enter Cluster Information.

IMPORTANT CLUSTER INFORMATION

In order for clustering to work, the name of the node must match the exact Windows Computer Name as it appears in the Windows System Control Panel. For example, if the Computer Name is REVNODE1, enter REVNODE1 as the Node Name. The name must be short with no spaces and cannot include a domain name.

Rev Node 1 IP Address: Rev Node 1 Name:

Rev Node 2 IP Address: Rev Node 2 Name:

InstallShield

< Back Next > Cancel

Caution: The Node Name must match the Microsoft Windows Computer Name as it appears in the Microsoft Windows System Control Panel.

The name must be short with no spaces and cannot include a domain name.

The name may contain numbers.

The name must be in UPPERCASE LETTERS.

23. Enter **Vbrick Rev Erlang Cluster** information. Choose whether this system is the **Primary** Vbrick Rev node and, optionally, enter a unique **Erlang Cookie Value** that will be the same across all nodes.

Vbrick Rev Erlang Cluster Information
Please enter Erlang Cluster Information.

Please choose whether this is the Primary Vbrick Rev node.

This is not the Primary Vbrick Rev node
 This is the Primary Vbrick Rev node

Please enter the Erlang Cookie Value.

NOTICE: This value must be the same across all nodes.

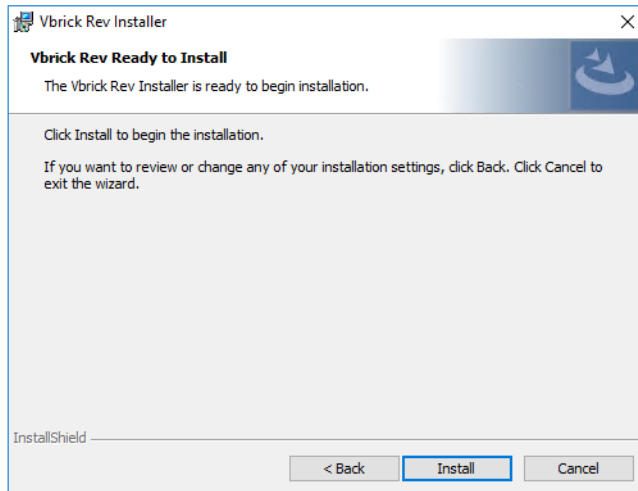
InstallShield

< Back Next > Cancel

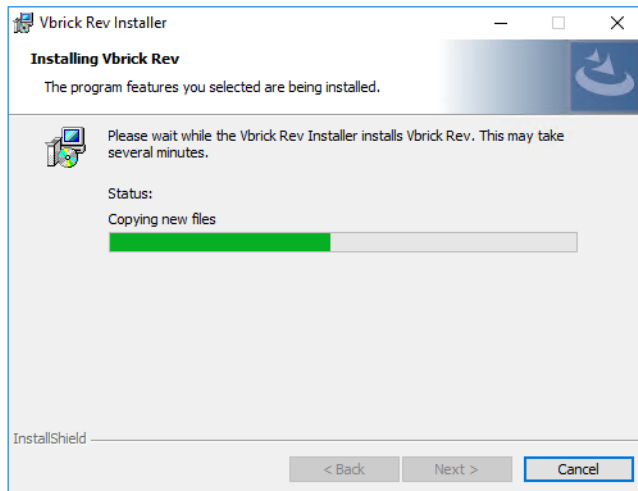
Note: Only one node can act as the **Primary** Vbrick Rev node. It is usually named REVNODE1.

The Erlang Cookie Value should be a string of at least sixteen characters and must be the same on all Vbrick Rev nodes. In most circumstances it is not necessary to change the default value **ERLANGcookieERLANGcookie** that the Vbrick Rev installer provides.

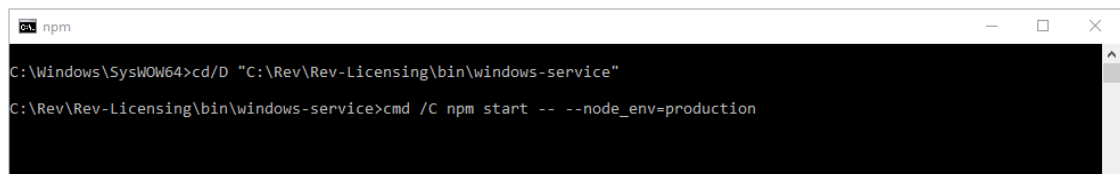
24. The Vbrick Rev Installer is now ready. Click **Install** to continue.



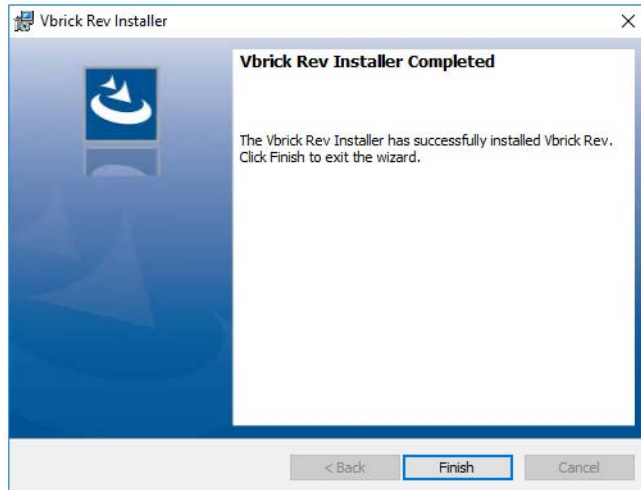
25. The Vbrick Rev Installer will take several minutes to install and configure Vbrick Rev.



26. The Vbrick Rev Installer will cause some Windows Console windows to appear and disappear. All of the windows will disappear automatically when the task is completed.



27. When the Vbrick Rev Installer is completed, the following window will appear. Click **Finish**.



28. Open a Windows Explorer window to `C:\Rev\VBrickPlatform.Runtime.Host.`
29. Choose one of the following scenarios.
 - For **Multi-Node** installations, right-click the file named `RabbitMQCluster.bat` and select **Run as administrator**.
 - For **Single-Node** installations, right-click the file named `SingleNode.bat` and select **Run as administrator**.
30. A window will appear that will proceed to set up the RabbitMQ clustering software. Press **Enter** to allow the process to proceed.

```

C:\Windows\System32\cmd.exe
RabbitMQCluster.bat:
RabbitMQCluster.bat: Welcome to VBrick Rev Multi-Node Configuration.
RabbitMQCluster.bat:
RabbitMQCluster.bat: Please ensure the VBrick Rev Service is not running.

Press any key to continue . . .
RabbitMQCluster.bat:
RabbitMQCluster.bat: Resetting local RabbitMQ.
RabbitMQCluster.bat:
RabbitMQCluster.bat: Error messages are normal and can be disregarded.

Stopping rabbit application on node rabbit@QAREV11E ...
Error: unable to perform an operation on node 'rabbit@QAREV11E'. Please see diagnostics information and suggestions below.

Most common reasons for this are:

```

31. When the process has completed, press any key to close the window.

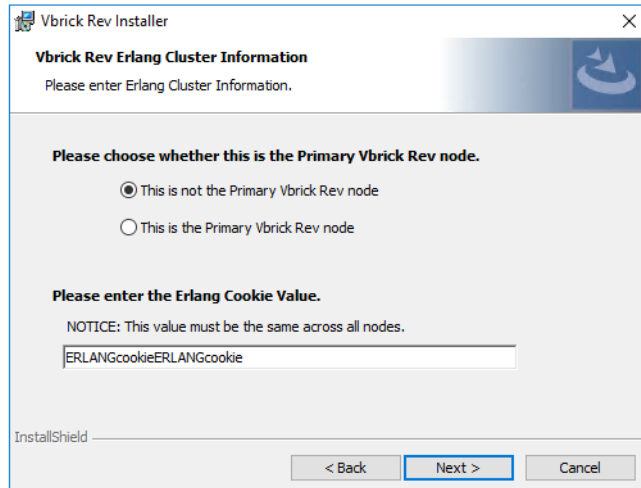
```

{[alarms, [{rabbit@QAREV11E, []}, {rabbit@QAREV11E, []}]]}
RabbitMQCluster.bat:
RabbitMQCluster.bat: You should have seen a status message
RabbitMQCluster.bat: describing the configuration.
RabbitMQCluster.bat:
RabbitMQCluster.bat: Script has finished.
RabbitMQCluster.bat:

Press any key to continue . . .

```

32. If this is the last Vbrick Rev node, continue to the [Configure Windows Firewall](#).
33. For **Multi-Node** installations, repeat these steps on the other Vbrick Rev nodes. On the Vbrick Rev Erlang Cluster Information window, ensure the other Vbrick Rev nodes are marked **This is not the Primary Vbrick Rev node**.



Configure Windows Firewall

Microsoft Windows Server has a restrictive Windows Firewall policy installed. Before checking connectivity, ensure the Windows Firewall is configured properly.

For enhanced security on Multi-Node installations, it is recommended to use the Windows Firewall built into Microsoft Windows Server.

Add Firewall Rules

External Vbrick Web Service

From an administrative command line on both REVNODE1 and REVNODE2 enable port 80:

```
netsh advfirewall firewall add rule name="VBrickRev80" protocol=TCP dir=in localport=80 action=allow
```

From an administrative command line on both REVNODE1 and REVNODE2 enable port 443:

```
netsh advfirewall firewall add rule name="VBrickRev443" protocol=TCP dir=in localport=443 action=allow
```

Internal Vbrick Rev Communications (Not required if single node)

From an administrative command line on REVNODE1 enable full communication between REVNODE1 and REVNODE2:

```
netsh advfirewall firewall add rule name="VBrickRevClusterIn" protocol=ANY dir=in action=allow remoteip=<revnode2-ip>
```

```
netsh advfirewall firewall add rule name="VBrickRevClusterOut" protocol=ANY dir=out action=allow remoteip=<revnode2-ip>
```


From an administrative command line on REVNODE2 enable full communication between REVNODE1 and REVNODE2:

```
netsh advfirewall firewall add rule name="VBrickRevClusterIn" protocol=ANY
dir=in action=allow remoteip=<revnode1-ip>
```

```
netsh advfirewall firewall add rule name="VBrickRevClusterOut" protocol=ANY
dir=out action=allow remoteip=<revnode1-ip>
```

Confirm Connectivity

Confirm that the Microsoft Windows Server 2012 node is able to reach the other Vbrick Rev node, the Elasticsearch Nodes 1 and 2, and MongoDB Nodes 1 and 2.

1. Please use the IP addresses of the nodes.
2. Open a command prompt and type:

```
ping <revnode1-ip>
ping <revnode2-ip>
ping <elastic-1>
ping <elastic-2>
ping <mongodb-1>
ping <mongodb-2>
```

Start Vbrick Rev Services

The steps performed in this section will need to be performed for each server you use in your installation.

1. On the **Primary** Vbrick Rev node, navigate to the **Windows Services Control Panel**.
2. Locate the following services. Right-click and choose **Start** on each one.

- Vbrick Rev Analytics Aggregator Service
- Vbrick Rev Analytics API Service
- Vbrick Rev Analytics Ingester Service
- Vbrick Rev Analytics Watcher Service
- Vbrick Rev Audit API Service
- Vbrick Rev Audit Service
- Vbrick Rev License Service
- Vbrick Rev Transcoding Fast Service
- Vbrick Rev Transcoding Slow Service

3. Once the **Vbrick Rev Analytics** servers are started, open a web browser and visit the URL **http://localhost:2222/version.js**. A message will appear that is similar to the following.

```
{sha: "6aa7ed64384afff2aa6300cb0b899736c1b4ada1", buildDate: "2021-04-20_02-05-15", buildNumber: "undefined.7",}
```

Caution: If you do not see this information, do not proceed with the installation. Contact Vbrick Support Services or your implementation team.

4. Once the **Vbrick Rev License Service** is started, verify that the license service is successfully running.
 - Open a web browser on the **Primary** Vbrick Rev node.

- Visit the URL `http://localhost:5555/v3/environment`.
 - You should see Rev licensing information on this page.
 - **If this URL does not load, follow the steps below and see if that fixes the issue.**
 - Open the `C:\Rev\Rev-Licensing\config` folder (or wherever you have Rev installed).
 - Make a copy of `production.json` and name it `default.json`.
 - Open up a Command Prompt or PowerShell as an administrator.
 - Type `cd C:\Rev\Rev-Licensing\bin\windows-service.`
 - Type `npm start. *`
 - Now open a web browser to the URL `http://localhost:5555/v3/environment`.
 - You should see the Vbrick Rev licensing information on this page.
5. If the licensing service fails to connect to MongoDB verify that the mongo "hosts" are set correctly in the licensing `production.json` file: `C:\Rev\Rev-Licensing\config`

Example:

```
C:\Rev\Rev-Licensing\config
{
  "mongo": {
    "hosts": ["QAMONGO01", "QAMONGO02"],

```

6. You should also verify that the license path and hostkey are correct.

```
},
"license": {
  "id": "Production",
  "path": "C:\\Rev\\license.lic",
  "hostKey": "123456"
},
```

Caution: If you still do not see the licensing information, do not proceed with the installation. Contact Vbrick Support Services or your implementation team.

7. Locate and select the **Vbrick Rev Service**. Right-click on the service and select **"Start."**
8. Once the services have started, check that all of the services whose names start with **"Vbrick"** are listed as **Running** in the Status column of the Windows Services Control Panel.
9. On the Vbrick Rev Nodes, these Vbrick processes can be seen in the **Windows Task Manager**.
- VbrickPlatform.Runtime.Host.exe
 - vbrickrevanalyticsaggregatorservice.exe
 - vbrickrevanalyticsapiservice.exe
 - vbrickrevanalyticsingesterservice.exe
 - vbrickrevanalyticswatcherservice.exe
 - vbrickrevlicenseservice.exe
 - vbrickrevtranscodingfastservice.exe
 - vbrickrevtranscodingslowservice.exe
10. Repeat the above process on the other Vbrick Rev nodes.

11. Once the VBrick Rev services have started, use a web browser and verify that you can successfully access the VBrick Rev Initial Configuration page. Access using the host name or IP of the HA Proxy / Load balancer.

Rev Audit Services Configuration - Installation

The audit service persists a “paper trail” of changes made to certain entities. It provides a detailed history of changes made to sensitive information, including who made the changes and when. This information is stored in a Mongo database called “rev_audit”.

By default the Rev Audit service is configured to automatically delete data older than 366 days from the audit database and the command to delete the data runs every hour.

To change these settings carry out the following steps on *all* Rev nodes where it was installed.

- Stop the Audit Service
- Edit the `olderThanDays` setting in the `C:\Rev\rev-audit\config\default.json` file and restart the Audit Service.

```
},
"server": {
  "purgeData": {
    "enabled": true,
    "olderThanDays": 366,
    "frequencyInHours": 1
  },
  "autoCreateIndexes": true
}
```

Verify Vbrick Rev Installation

To verify your Vbrick Rev Runtime Service installation, use a web browser to visit the IP address or Hostname of each node running Vbrick Rev.

You are now ready for the [Post Installation Review](#).



Post Installation Review

The purpose of this document is to guide a Vbrick Rev System Administrator through the process of initial steps to take once Rev installation is complete. This includes enabling Guest Access (if applicable) and the Rev Root Account configuration which occurs just after installation.

Configure the Rev Root Account

Last Updated: November 13, 2018

Using a web browser open the IP address or the Hostname of the HA Proxy.

The first screen to appear once your installation is complete is the Root Account configuration screen.

Caution: Be aware that the **Root Account** contains certain permissions that no other account will contain. *Remember the user name and password to this account.*

Once the installation of the Vbrick Rev Service is complete, enter the address of one of the Vbrick Rev nodes in a web browser to access the system.

Initial Configuration

Step 1 of 2 Enter Contact Information

ROOT ACCOUNT INFORMATION

Account Name: *

Account Host Name:

TIMEZONE: ▼

ROOT USER INFORMATION

Username: *

Password: *

Confirm Password: *

Email: *

First Name:

Last Name: *

Title:

Phone Number:

EMAIL SERVER INFORMATION

Email Server Address: Port:

Email Server User Name:

Email Server Password:

Email 'From' Address:

Enter Contact Information

Enter the Root Account information:

- Enter the **Root Account** name. This should be considered the parent company name.
- Enter the **Host Name** or the **IP Address**.

Initial Configuration

Step 1 of 2 Enter Contact Information

ROOT ACCOUNT INFORMATION

Account Name:

Account Host Name:

TIMEZONE: ▼

Enter the Root user information:

- Enter a **Username** and **Password** for the root user.
- Confirm the **Password**.

- Enter the root user's **Email, First Name, Last Name, Title, and Phone Number.**

ROOT USER INFORMATION

Username:

Password:

Confirm Password:

Email:

First Name:

Last Name:

Title:

Phone Number:

Caution: This is the Master Root Account Admin, or **Super User**, for the Vbrick Rev system. Once this account is created it is **highly** recommended that another Admin account be created with all permissions.

The **Root Account** should *not* be used for daily tasks. If the Root Account is locked out, and there is not another Admin account with Admin access, there is **no** way to unlock the Root account.

Required fields are identified with the red asterisk (*).

Enter the email server information:

- Enter the **Email Server Address** and **Port** number.
- Enter the **Email Server User Name** and **Email Server Password**.
- Enter the **Email 'From' Address**. When users receive email from the Rev System, this is the email address that will be displayed to them in the 'from' field.
- Click **Enter Contact Information** to continue.

EMAIL SERVER INFORMATION

Email Server Address: Port:

Email Server User Name:

Email Server Password:

Email 'From' Address:

[Enter Contact Information](#)

Enter additional Contact Information and the Billing Address information:

- Enter a **First Name** and **Last Name**.
- Enter additional required contact information.
- If the Billing Address is different than the Contact Information, click **Enter a different billing address** checkbox.

Initial Configuration: root

Step 2 of 2 ◀ Back Create Root Account

CONTACT INFORMATION

First Name:

Last Name:

Contact Email:

Address Line 1:

Address Line 2:

Country: ▼

State:

City:

Postal Code:

Phone Number:

Preferred Language: ▼

Enter a different billing address

◀ Back **Create Root Account**

- When Contact and Billing details are complete, click **Create Root Account**.
- The system will initialize your environment and allow you to login.

Login as Root User


- Once initialized, the **Sign In** screen will appear.
- Enter the Root **Username** and **Password** you just created.
- Click **Sign In**. You are now logged in as the Root User Account.

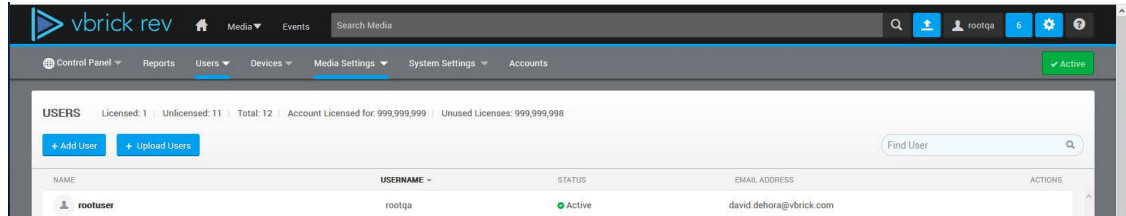
Sign In

(Forgot Password?) **Sign In**

Create a Rev Admin Account

As previously mentioned, you do *not* want to use the Root User Account for day-to-day functions in Rev. Instead, create an Admin Account for this.

- Click the **Gear** icon  on the Header navigation menu to access **Admin** menu functions.
- Click the **Users** dropdown menu.
- Click **Add User**.



On the **New User** screen, enter the user information:

- First name
- Last name
- Contact email
- Title
- Phone number
- Preferred language
- Username

Click the plus sign (+) for the next sections (**Role Assignment and Group Assignment**) if not already expanded.

- In the **Role Assignment** section, click a role from the right list (**Available Roles**) to add to the **Assigned Roles** on the left.
- To remove all assigned roles, click the **Remove All** button.
- To add all available roles, click the **Add All** button.
- In the **Group Assignment** section, click a group from the right list (**Available Groups**) to add to the **Assigned Groups** on the left.
- To remove all assigned groups, click the **Remove All** button.
- To add all available groups, click the **Add All** button.

Click **Save** to create the new Rev Account Admin user.

Troubleshooting the Installation

Last Updated: November 13, 2018

This section describes how to troubleshoot the Elasticsearch, MongoDB, and Vbrick Rev systems.

Elasticsearch Troubleshooting

Caution: Elasticsearch Cluster status may remain “**red**” for up to ten minutes after installation has completed. Please allow time before proceeding to the Troubleshooting section.

Single-node Elasticsearch installations usually do not reach higher than “**yellow**” status.

There are some specifics to keep in mind regarding the Elasticsearch service:

- Vbrick Rev uses Elasticsearch as a data store. Make sure you still have free disk space.
- Generally speaking, Elasticsearch is a very resilient service so the primary concern is making sure the process is running on each node.
- Elasticsearch log files are often quite verbose, and more so when there are issues with the cluster so after making sure the process is running on all Elasticsearch instances, always look to the logs.

In **Elasticsearch Installation**, we provided a curl command for validating your installation. This command also useful for checking the status of Elasticsearch during troubleshooting and there are additional curl commands that may come in handy.

Cluster Members

Testing individual health is useful, but it's also good to know what the cluster thinks. Use this command on any node to get a list of all cluster members.

```
Curl
curl -XGET 'http://localhost:9200/_cat/nodes?v'
```

The output of most Elasticsearch queries is in JSON, however some responses are in formatted, plain text. The output of the previous command should look something like this:

Output

```
ip          heap.percent ram.percent cpu load_1m load_5m load_15m node.role master
name
10.10.2.13  26           82    0    0.00    0.00    0.00 mdi      *      ES-
02
10.10.2.15  24           93    0    0.00    0.00    0.00 mi       -      ES-
03
10.10.2.12  39           74    0    0.17    0.05    0.01 mdi       -      ES-
01
```

We have three servers in this example, two data nodes and one arbiter. Items of note:

- The asterisk in the master column designates the master node of the cluster.
- The `node.role` value can be one of three: `m` - master capable node, `d` - data node, `i` - ingest node
- The `cpu` and `1/5/15` load numbers should all reflect accurate system values for each node however under heavy load the values may not be up to date
- The `ip` and `name` should match what you expect. Elasticsearch is dependent on hostname resolution and the Vbrick Package Installer makes `/etc/hosts` modifications to insure hostname lookup is concrete on all nodes. If something on the node has changed, the values in the hosts file may no longer apply. Make sure your networking and hosts file match.

Wipe Elasticsearch Cluster

Caution: Analytics and search data is *permanently* erased when performing this step on your Elasticsearch instance and cannot be recovered. Search data can be re-indexed; analytics data cannot. Please ensure that this is the appropriate course of action before proceeding with this process.

In some circumstances it may be necessary to wipe an Elasticsearch instance and start over. Indexing commands like PUT and DELETE execute across a cluster of machines so it doesn't matter which node you're attached to when you execute a command, just as long as the process is running on the node you're using you will be able to reset y Analytics and search data will be permanently erased when performing this step and cannot be recovered.

Use the curl command below to wipe the Elasticsearch cluster.

Curl

```
curl -XDELETE 'http://localhost:9200/rev'
```

MongoDB Troubleshooting

Most issues with MongoDB arise when clustering is involved. Single-node instances of MongoDB lack the complication of clusters and as a result most of the troubleshooting steps involve clusters.

There are some specifics to keep in mind regarding the MongoDB service:

- Vbrick Rev uses MongoDB as a data store. Make sure you still have free disk space.
- The MongoDB driver used by Vbrick Rev is resilient and can often work through issues when MongoDB slaves become unavailable. Make sure you have another way to monitor MongoDB health in addition to Vbrick Rev.
- MongoDB replication uses master/slave topology. Once you have determined your acting-master MongoDB node [use `rs.status()`] make sure that node is healthy and the command line client is responsive.

Resetting a Slave

Under some circumstances, a slave MongoDB instance can lose track of its master. Look for the `lastHeartbeatMessage` in the output of `rs.status()` - if you see **could not find member to sync from** that member has lost sync with its master. The process as suggested by the authors of MongoDB is a simple reset of the slave, but it is *critical* you execute this process only on the affected slave because the reset is total - all data on the slave is removed and rebuilt.

The following commands would reset a known MongoDB slave running on Ubuntu 14.04:

Reset MongoDB Slave

```
service mongod stop
cd /var/lib/mongodb
rm -rf .//*
service mongod start
```

Note: Data on Red Hat systems is in `/var/lib/mongo`

When the MongoDB service restarts with an empty directory, it will immediately reach out to the master and re-sync. You can also back up the MongoDB data directory before you remove all the files from it - the backup would suffice to rollback to before the re-sync, however the node would return to its un-sync'd state. Resetting a slave in this way is a best practice, even if it does seem a little drastic.

Manually Clustering MongoDB

There should not be any reason to re-cluster MongoDB unless networking information has changed. MongoDB is reliant on the quality of the data entered for replication so the Vbrick Package Installer uses entries in `/etc/hosts` to make concrete the networking information for the MongoDB cluster and other nodes in the Rev solution. If any of this information has changed, it may be the source of instability so it's good to make sure networking and host file entries still agree. Nodes in MongoDB clusters can be added or reassigned, but this is a manual process outside of the scope of this documentation.

Wipe MongoDB Cluster

Caution: Data is *permanently* erased when performing this step on your VBrick MongoDB Cluster. Please ensure that this is the appropriate course of action before attempting this process.

Please contact Vbrick Support for assistance when data preservation is required.

In some circumstances it may be necessary to wipe the MongoDB Cluster and start over. Data is *permanently* erased when performing this step and recovery will require some form of previous backup. Make sure you've logged into your master MongoDB instance (if this is a cluster) and use `sudo mongo` to enter the MongoDB client; use **ctrl+d** at any time to exit.

Once at the prompt, use the following commands to wipe the MongoDB cluster.

Wipe MongoDB Cluster

```
use rev
db.dropDatabase()
use rev
use rev_licensing
db.dropDatabase()
use rev_licensing
```

Make certain not to skip the secondary use - in MongoDB, typing use **<databasename>** on an empty database will create it so this line cannot be skipped.

Log into a Windows node and open a command prompt.

Type the following commands:

```
telnet 127.0.0.1 5555
PUT /v3/load HTTP/1.0
```

Press the **Enter** key to terminate that last line and then hit **Enter** a second time. You will see a message that looks similar to the following:

```
[{"name": "Rev_Active_User_Seat", "acquired": 0, "available": 10, "total": 10, "unlimited": false}, {"name": "Rev_Server", "acquired": 0, "available": 1000, "total": 1000, "unlimited": true}]
```

Caution: Do not drop the **Admin** database. If the Admin database is lost, MongoDB will require manual configuration or you will need to begin the MongoDB installation with a fresh device/instance.

Rev Runtime Service Troubleshooting

Start Rev from the Console

For troubleshooting purposes you may choose to start the **Vbrick Rev Service** in the Windows Console and observe log messages.

1. Open the Start Screen and type `cmd`.
2. Right-click on the command prompt and select **Run as administrator**. In the command window that opens, enter the following commands.

```
cd C:\Rev\VBrickPlatform.Runtime.Host
.\VBrickPlatform.Runtime.Host.exe /console
```

To get a better view of console logs you may choose to enlarge the “Command Prompt” window.

Go to the “**Defaults**” menu on the “Command Prompt” window and increase the **Width** and **Height** of the Window Size. Do the same thing with the “**Properties**” menu on the “Command Prompt” window.

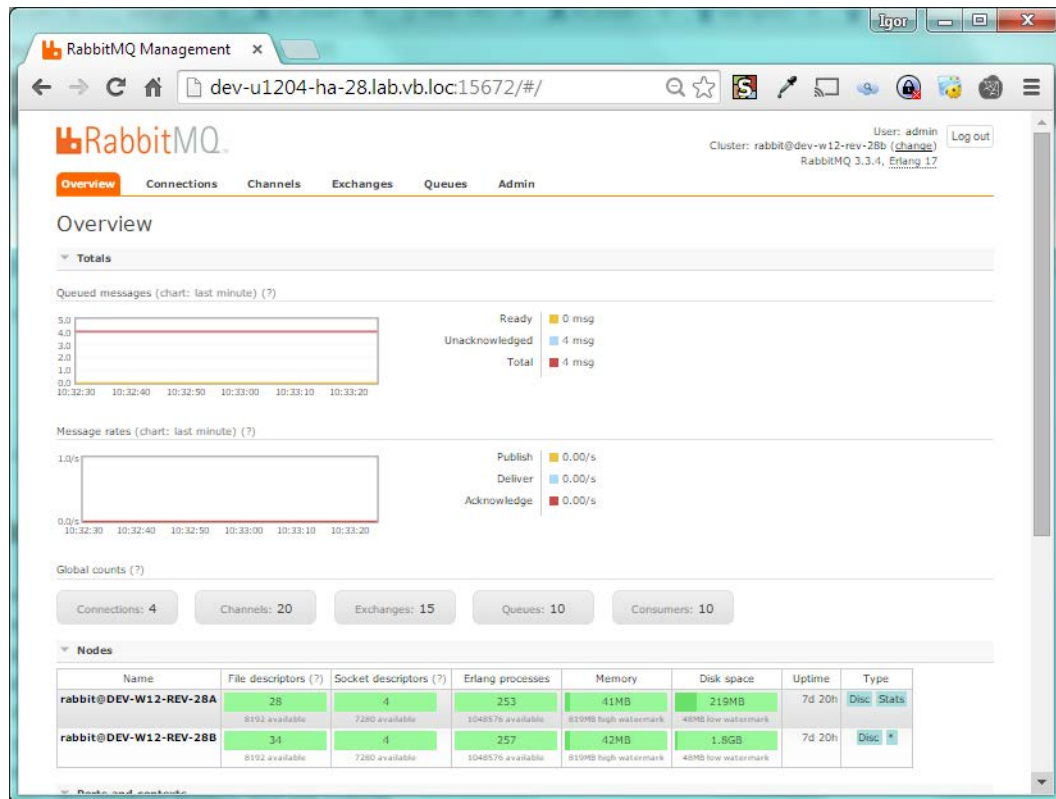
RabbitMQ

Management Web Interface

RabbitMQ has an optional management plug-in that is enabled by default once you've completed the Rev installation by running either `SingleNode.bat` or `RabbitMQCluster.bat` on the Windows nodes. The management plug-in adds little additional overhead to the message queue and provides web-based management and monitoring of the RabbitMQ cluster.

- Visit <http://127.0.0.1:15672> and log in with user **admin** and password **admin**.

Although this screen is using HAProxy to connect to RabbitMQ management, your screen will look very much the same:



Cluster Rebuild

It is possible to rebuild the RabbitMQ Cluster at any time without destructively affecting the Vbrick Rev service.

1. Log-in to Vbrick Rev **Node 1**.
2. Open the Services Control Panel and locate all of the Vbrick services.
3. Stop *all* Vbrick services.
4. Close the Services Control Panel window (this is necessary or this process may fail).
5. Open Windows Explorer and navigate to `C:\Rev\VBrickPlatform.Runtime.Host`
6. Locate the file named **RabbitMQCluster.bat**
7. Right-click the file named RabbitMQCluster.bat and choose **Run as administrator**
8. The system will guide you through a series of prompts that rebuild the cluster and will exit.
9. Log into Vbrick Rev **Node 2**.
10. Repeat steps 2 through 8.
11. Log into Vbrick Rev Node 1 and start all the Vbrick services, with VBrick Rev Service being started last.
12. Log into Vbrick Rev Node 2 and start all the Vbrick services, with VBrick Rev Service being started last.
13. Visit `http://127.0.0.1:15672` with a web browser.
14. Log in as user **admin** and password **admin**.
15. Note there are two nodes with the correct names in the cluster.

This process can be especially helpful if the Vbrick Rev Nodes have Windows Computer names that do not match the Rev Node Names provided to the installer. Look into the

`C:\Windows\System32\Drivers\etc\hosts` to see what the Windows Computer names should be, change them in the Windows System Control Panel, restart the computer, and follow the instructions above to rebuild the RabbitMQ Cluster.

Caution: The Windows Computer name *must* match the Windows host name as noted in the `C:\Windows\System32\drivers\etc\hosts` file. This file should have already been set up for you by the Vbrick Rev Installer.

The Windows Computer name must be in **ALL UPPERCASE LETTERS**.

HAProxy and Load Balancer Troubleshooting

Troubleshooting an HAProxy is recommended for advanced users only.

1. Log into the HAProxy node and check that all nodes are in `/etc/hosts` and are reachable.
2. View the HAProxy logs at `/var/log/haproxy.log` for warnings and errors.
3. Using information from the logs, check the config file at `/etc/haproxy/haproxy.cfg`
 - All front-end elements must have corresponding backend elements that contain reachable servers.
 - The HAProxy node itself can reach all of these
 - If HTTPS is required, verify that the SSL information in `/etc/haproxy` is valid



Appendices

The various appendices included here may be useful for additional information or instructions during an installation or upgrade.

Appendix A: General Information

Last Updated: June 3, 2021

This document provides additional details supporting an installation and configuration of Vbrick Rev.

SSH to a Linux Server Using PuTTY on Windows

1. On Windows, download the latest PuTTY from <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>.
2. Accept SSH fingerprint.
3. Enter the username and password and you should be connected to the host.

SSH to a Linux Server Using Terminal and SSH

Linux and Mac OSX both have built-in SSH clients.

- For Mac OSX: use the `Cmd + Space` keys to open **Spotlight**, and type in terminal - hit **Enter** when you see an icon appear. The Apple Terminal application lives in the in Applications/Utilities folder and you can lock it to the Dock if you plan to use it often.
- For Linux: if you are using GUI, there are numerous terminals depending on your window manager. Applications like xterm or Gnome terminal and numerous others are all suitable. Anything that provides access to a shell on the local system will do, including a direct console.

With your terminal window open, complete the following steps:

1. Open a (remote) secure shell using the command: `ssh user@hostname`
2. If this is the first time you are connecting to the remote host, answer `yes` (you must type the full word) when asked if you should accept the SSH key.
3. Enter your password if prompted. SSH configuration allows for access using private/public key pairs that must be set up in advance to take advantage of this feature. SSH public/private key setup is beyond the scope of this documentation however we mention it here because if it is in use within your organization, you may not be prompted for a password.

Copy File to a Linux Server Using PuTTY PSCP on Windows

1. Download the latest PuTTY PSCP from <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>.
2. Unblock the exe file, right-click on the file go to **Properties**, and click **Unblock**. If the **Unblock** button doesn't show up, then it is already unblocked.

-
3. In a terminal window, CD to the directory you saved the pscp binary, then run `pscp user@host`
 4. If prompted to accept the ssh key, type `y` for yes (this only happens the first time you connect to a new host).
 5. You will be prompted for a password, after entering, the file will be copied.

Copy a File to Linux Server Using WinSCP on Windows

1. Download the portable WinSCP executable from <http://winscp.net/eng/download.php>.
2. Unzip and Unblock the exe file.
3. Create a **New Site** with the following attributes:
 - Protocol: SFTP
 - Host Name: riak.local
 - Username: Vbrick
 - Password: XXXX
 - Then click **Login**.
4. Choose **Yes** to accept the host key.
5. Highlight file on the left plane.
6. Drag File to the right hand Plane and choose **OK** at the prompt.
7. You will now see the file on the right hand plane.

Copy File to a Linux Server Using a UNIX Machine

1. Open a terminal, on MAC OSX, its in the in Applications/utilities folder, other Unix/Linux variants will be different.
2. Run `scp file user@hostname:`
3. If prompted to accept the ssh key, type `yes` (this only happens the first time you connect to a new host).
4. The file has now been copied to the remote machine.

Appendix B: File Storage Path Configuration

Last Updated: May 15, 2020

This document describes the file storage folder options available with Vbrick Rev and how to configure them. Keep in mind this refers to On-Premises installations only.

Important Notes:

If you are using [Windows Server 2019](#), please make sure you review that specific section in this document for new behaviors.

Caution: This section refers to On-Premises installations with unusual storage requirements *only* and is *not* intended to be used by Rev Cloud installations or most Rev customers.
Contact Vbrick Support if you have any questions.

Rev File Storage

Any type of file that is uploaded to Vbrick Rev needs to be stored on disk. This list includes:

- Videos and all transcoded instances of a video
- Thumbnails

- Presentations
- Supplemental files that are attached to a video

Even though the Vbrick Rev Installer refers to this option as **LocalFolder**, the folder doesn't need to be local. Any suitable filesystem URL that describes a Rev-compatible store can be used as the value for **LocalFolder**.

The File Storage Path configuration options that may be used with Vbrick Rev are as follows:

- Any path with a drive letter. This includes paths similar to `C:\Rev\Rev.FileStore` or `D:\<some_path>` or any mapped drive letter, such as `Z:\<some_path>`
- A Windows Universal Naming Convention (UNC) Path, such as `\\SERVERNAME\FolderName`

Note: Mapped Drives

Drive letters for mapped drives are not necessarily the same on every system, make sure you confirm the path(s) on each system where you run the Windows Rev Installer.

Caution: Remote file systems will require user access for the Windows Service Account used during setup. Make sure the Windows Service Account has read and write access to any non-local filesystem.

Remote File System Considerations

When working with **N+1** Rev Runtime nodes, some form of shared filesystem is a requirement. The Rev Runtime service does not synchronize files between nodes and instead relies on a consistent file store that has everything placed there by any running Rev Runtime node. There is no fixed requirement for how one might implement this shared file system but there are some factors that should be considered.

- The preferred solution is a resilient, remote file system supported by a server or device separate from any of the Rev Runtime nodes. This device can include Network Attached Storage (NAS) and redundant Storage Area Networks (SAN). Storage of this type will provide additional redundancy via underlying RAID or other mechanisms in addition to easy expansion apart from the main nodes.
- A simple file share can also be used but the structure behind that share must be carefully considered:
 - You should not use either of the Rev Runtime nodes as a filesystem source for the other or both. A configuration of this type will not provide redundancy in most failure circumstances.
 - The underlying filesystem for the share will affect user experience. Multi-disk RAID is encouraged.
- Any system, redundant or otherwise should be backed up. Larger storage systems often facilitate snapshot backups which don't interfere with production use, but simpler types of full volume backups to an external device or cold storage is strongly suggested.

Configuring Windows Paths for LocalFolder

There are a number of supported options for filesystem path support in Rev Runtime.

Drive Letters

Any Windows path that starts with a drive letter can be used.

-
- C:\Data\RevData
 - D:\Rev.Data
 - Z:\Rev\Data

In the last example, drive letters starting at the end of the alphabet are usually reserved for mapped network drives. Make sure this drive letter is the same or make note of and use the difference when you're configuring a letter at the end of the alphabet.

Universal Naming Convention (UNC) Paths

A Universal Naming Convention path starts with two back-slashes, and names some resource that is intended to be universal on that system or network. A typical UNC path looks like this `\\<some_resource>\path` and the resource can be a server, printer or some other device available for wide access. UNC paths tend to be consistent within a network so a path used on one Rev node will probably be the same for any other nodes.

NFS and Other Remote Filesystems

Rev can support any remote file system that Windows supports provided the Windows Service Account that executes Rev Runtime has read and write access to that filesystem. Configuring these filesystems is outside of the scope of this document however, it should be made clear that the limitations in this regard are few. As long as the filesystem is available and performant, Rev can make use of it. There are a couple of factors to keep in mind when setting up a filesystem of this type for Rev:

- Make sure that system-to-system mount locations are the same. Rev nodes don't exchange configuration information so while you could have each machine with a different mount point and have them configured correctly, we feel this only adds additional complication for upgrades and the diagnoses of problems.
- Always test your storage accessibility before you begin the installation process.

Change an Existing Storage Path

File Storage Path Configuration Overview

Before You Begin:

All prior Vbrick Rev installation procedures and steps must be complete before you configure your file storage path.

You will need to understand or obtain the following information:

- **Vbrick Rev Service** - This file is accessed, started, and stopped during the configuration process through the **Microsoft Windows Services Control Panel**
- **Vbrick Rev Configuration File** - You will need to access and edit this config file as an administrator
 - This file can be found in `C:\Rev\VBrickPlatform.Runtime.Host\VBrickPlatform.Runtime.Host.exe.config`
- **UNC Path** - A Microsoft Windows Universal Naming Convention (UNC) path used to connect with a remote File Storage folder such as: `\\SERVERNAME\FolderName`

File Storage Path Configuration Step-by-Step Overview

To select and configure a file storage path for Vbrick Rev, complete the following steps:

1. Stop the Vbrick Rev Service.
2. Edit the **Vbrick Rev Configuration File** to configure a file storage path.

3. Configure the Vbrick Rev Service to run as a Microsoft Windows User that has permission to access the file storage path.
4. Start the Vbrick Rev Service.

Stop the Vbrick Rev Service

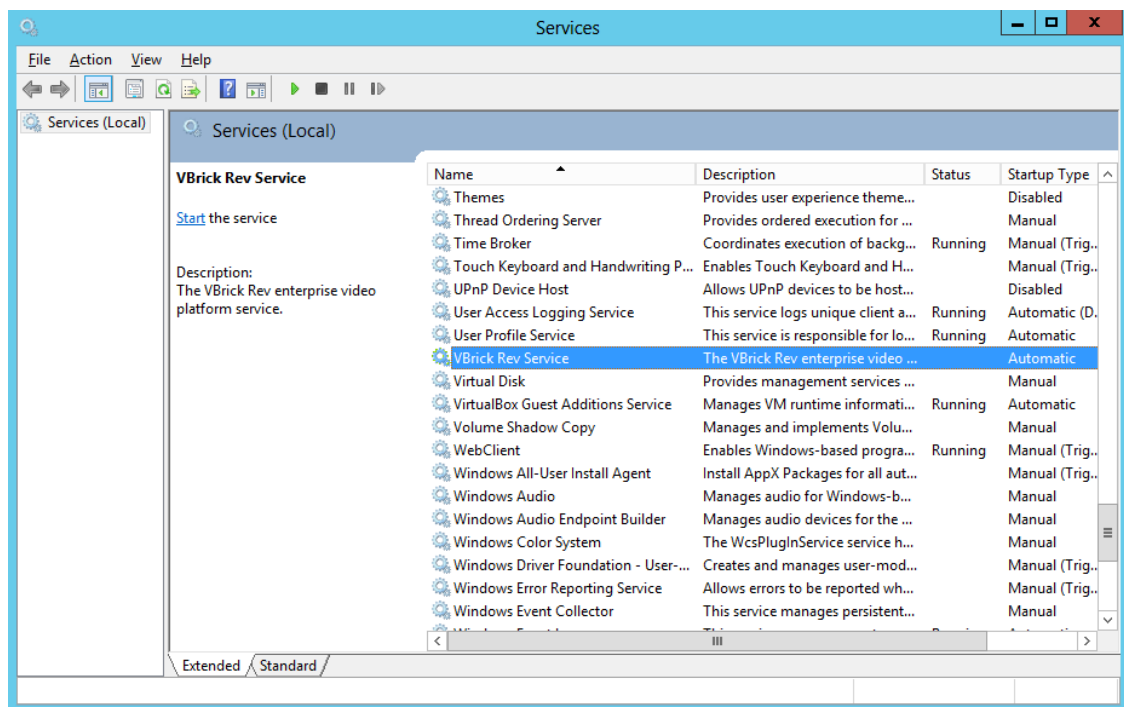
Before you begin, you must stop the Vbrick Rev Service.

Note: Stopping the Vbrick Rev Service will disconnect any users currently using Vbrick Rev.

Caution: If you change the File Storage Path, Rev will lose access to data stored in the old location. Always establish the File Storage Path before you begin populating your Vbrick Rev system with data.

To stop the Vbrick Rev Service:

1. Use Windows Services (Extended tab), seen below, and scroll down to the **Vbrick Rev Service**.
2. Highlight the service and click the **Stop the service** hyperlink to the left.
3. You may also right click on the service and click **Stop**.



Edit the Rev Configuration File to Configure a File Storage Path

After the Vbrick Rev Service has been stopped, you must edit the **Vbrick Rev Configuration File** to reflect your selected storage path configuration.

The config file may be found here:

```
C:\Rev\VBrickPlatform.Runtime.Host\VBrickPlatform.Runtime.Host.exe.config
```

Always edit the configuration file as an administrator in Notepad.

LocalFolder: Folder with a Drive Letter

The Vbrick Rev Installer will likely have already configured this for you.

```
language: xml
<remoteFileStorage type="LocalFolder">
  <localFolder path="C:\Rev\Rev.FileStore" />
</remoteFileStorage>
```

LocalFolder: Mapped Network Drive Letter

The Vbrick Rev Installer will likely have already configured this for you.

```
<remoteFileStorage type="LocalFolder">
  <localFolder path="Z:\" />
</remoteFileStorage>
```

This storage path may be configured a number of different ways in Microsoft Windows. Consider this example:

```
language: powershell
NET USE Z: \\SERVERNAME\FolderName password /USER:username /SAVECRED
NET USE /PERSISTENT:yes
```

LocalFolder: Windows Universal Naming Convention (UNC) Path

The Vbrick Rev Installer will likely have already configured this for you.

```
language: xml
<remoteFileStorage type="LocalFolder">
  <localFolder path="\\SERVERNAME\FolderName" />
</remoteFileStorage>
```

This storage path may be configured a number of different ways in Microsoft Windows. Consider this example:

```
NET USE \\SERVERNAME\FolderName password /USER:username /SAVECRED
NET USE /PERSISTENT:yes
```

Configure Vbrick Rev Service as a User with Access to the Storage Path

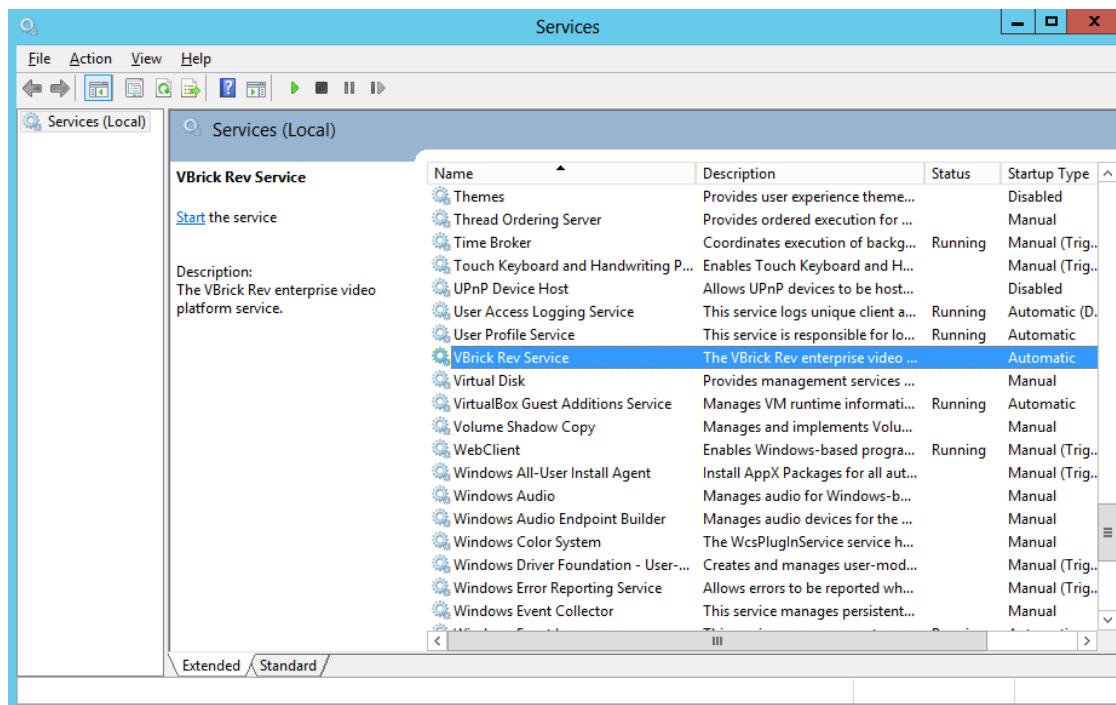
This step must be completed if you are using the **LocalFolder: Mapped Network Drive Letter** or the **Windows Universal Naming Convention (UNC) Path**. If you are using any other type of configuration, you may skip directly to restarting the Vbrick Runtime Host service.

These settings will be removed when Vbrick Rev is upgraded. If you are upgrading, this process in this document must be re-applied.

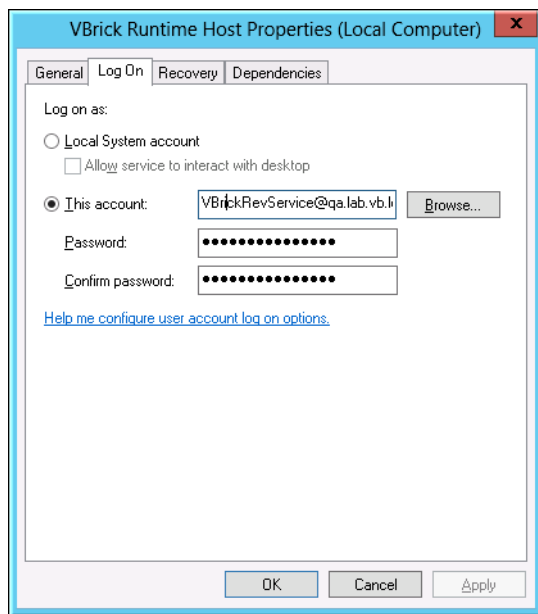
Once the config file has been edited, the Vbrick Rev Service needs to be configured to run as a user with access to the storage path you just set. This step only needs to be done once.

To start the Vbrick Rev Service as a User with Access to your Storage Location:

1. Log in to your Vbrick Rev Service nodes as an Administrator user and open the **Services** window.
2. Select the **Vbrick Rev Service**.



3. Right click on the **Vbrick Rev Service** and select **Properties**.
4. Select the **Log On** tab and enter the following information:
 - **This account:** Select **Browse** and locate a User Account that has permission to access the File Storage Path selected earlier
 - **Password:** Enter the password of this User Account
 - **Confirm password:** Enter the password of this User Account again to verify.
5. Click **OK** to save your information.



Restart the Vbrick Rev Service

The final step is to restart the Vbrick Rev Service service.

To start the Vbrick Rev Service:

1. Use Windows Services (Extended tab) and scroll down to the Vbrick Rev Service.
2. Highlight the service and click the Start the service hyperlink to the left.
3. You may also right click on the service and click Start.
4. You are now ready to configure and use your Rev installation.

Windows Server 2019

The Common Internet File System or CIFS is the current iteration of “Windows File Shares”. The underlying protocols have always been a default in Windows Servers so much so that the Linux world has had to implement various solutions to cope with it.

Starting with Windows Server 2019, CIFS is no longer enabled by default and this includes both server and client processes. Enabling CIFS is a relatively simple and straight forward task however if this step is skipped you will run into issues when trying to mount remote shares on Windows Server 2019.

Appendix C: Backup Elasticsearch and MongoDB

Last Updated: December 06, 2018

This document describes how backups are managed for Elasticsearch and MongoDB. Rev stores mission critical data in both data stores. It is therefore also critical for an organizational plan for regular backups outside of the maintenance window that an upgrade might provide.

Core Concepts

If there were to be one core enterprise mantra, it should be that backups are important. The degree of importance is usually debatable right up until it's found that the backups either aren't available, don't work, or don't contain the required business data. Setting up a plan and methodology in advance will keep all failures that involve that data to a manageable degree of severity.

Regular Backups

Each organization should maintain a regular backup schedule that matches their data use and business continuity plan. The On Premises release of Rev does not provide automated, regular backups for either Elasticsearch or MongoDB although both methods are documented here and here.

Maintenance Backups

Vbrick strongly suggests backing up your data before any upgrade. The On-Premises installer gives you the opportunity to do this during the upgrade process but you also have the option of using any processes developed by your organization to gain similar redundancy and rollback data before you proceed with your upgrade.

VM Snapshots

If you are using virtualization for your various hosts, Vbrick strongly suggests taking advantage of this mechanism to help maintain data and node resilience.

Dependencies

Caution: Any time you are taking a backup of either or both data sources, make certain all Windows Rev services on all Windows Rev nodes are shut down so that all data operations have ceased before proceeding with your backups.

Free disk space is the leading cause of failure for most all backup endeavors. In most cases there is either not enough space to begin a backup profile or over time the space runs out because there was no warehousing or cleanup plan for the backup data. Always ensure enough disk space is available on each node before you begin your backup process and again after the backup process to make sure that you can maintain your backup plan over a long term.

Automated Backups

Backups can be handled quite easily as part of the upgrade process. After node access is established, content is prepared and Rev has been verified to be shut down, you are given the opportunity to let `vbrick-setup` backup your data.

```
Do you wish to backup Elasticsearch? (yes or no)
yes

Do you wish to backup MongoDB? (yes or no)
yes

Backing up each data-bearing Elasticsearch node
- [qaes01] Elasticsearch backup complete: ✓
- [qaes02] Elasticsearch backup complete: ✓

Backing up MongoDB: ✓
```

If you select yes for either Elasticsearch or MongoDB, `vbrick-setup` will then backup the appropriate data. For Elasticsearch, `vbrick-setup` backs up each data-bearing node individually. For MongoDB, `mongodump` is used to gather a working copy of the data from the working set of MongoDB servers. This operation is performed on the first server in the set, regardless of which node is the current master.

Backup Locations

Backups will be stored in specific locations on the data bearing nodes.

Caution: If the data directory (usually `/var/lib/<service>`) for the service being backed up shares the same partition as `/var/lib` - you must make certain to regularly inspect and maintain the free space on this partition. Accumulated backups may prevent the upgrade process.

Elasticsearch

Each Elasticsearch server that carries data (Arbiters are excluded) is backed up in a methodology similar to the one documented below. The location of the backups is predetermined by the automated process and the resulting gzip-compressed tar-files can be found in `/var/lib/elasticdump`. Each backup, if used to restore data must be used on the node where the files were created and should not be used on any of the other nodes.

MongoDB

The MongoDB backup process backs up the entire cluster using only the first server in the list of MongoDB servers. Because the process uses `mongodump`, the output a complete set of utility-compatible binary files for restoring the database. These files are archived using tar and compressed using gzip into a single archive and placed in `/var/lib/mongodump`. If you choose to restore data from these backups, choose the archive file with the data suiting your needs, and then follow the restoration process documented in the next section at [Manual Backups > MongoDB > Restore](#).

Manual Backups

If you don't intend to take advantage of the automated backup process during upgrade or you want to provide your own backups under any circumstance, the following manual methods are available.

Elasticsearch

You can use one of two methods to back up Elasticsearch instances:

- Back up the filesystem
- Let Elasticsearch manage the process via Snapshot and Restore

The process described below is a filesystem backup method. Managing Elasticsearch snapshots is beyond the scope of this documentation, more information about that process can be found [here](#).

Backup

The Elasticsearch process has the Apache Lucene search engine at its core. The data files that Lucene generates are safe to archive and move around, provided the process is not running when the archiving takes place. Backing up Elasticsearch at the file level is very straightforward and reliable. There are a couple of caveats when using this backup method:

- You *must* shut down the Elasticsearch process on each server that is running the service (this includes Arbiters).
- All the servers in your Elasticsearch instance must be fully backed up before restarting the process.
- The Elasticsearch data is tied to the hostname that generated it. If you've taken a backup on a host named `host-a`, Elasticsearch will have issues consuming the same data on a host named `host-b`. Make certain that if you move your backups to off server locations that you have some convention for ensuring those files return to either the hosts that created the backups or hosts that are named the same.

Log in as the user `vbrick` on the primary Elasticsearch node and escalate your privileges with `sudo su -`

Elasticsearch data is kept in `/var/lib/elasticsearch` - you should first check the amount of space currently being consumed in this location and then ensure there is suitable space elsewhere (or on the same partition) for the backup. The following two commands will assist in this process.

```
df -h
```

This command will display the mounted filesystems currently hosted on this node. The `-h` toggles human-readable format so that the space on disk is easily calculated.

```
du -h --max-depth=1 /var/lib/elasticsearch
```

This command will display the specific amount of space being used in the Elasticsearch data path.

Once you're confident about where you will place your backups, stop the Elasticsearch service on *all* the Elasticsearch nodes.

```
service elasticsearch stop
```

Check to confirm the process java has stopped running.

```
ps -ef | grep java
```

With the Elasticsearch process fully shutdown, you can safely tar the data directory to your chosen path, in this example <target path>. Our example also uses a crude timestamp that should reflect your current circumstances.

```
cd /var/lib
tar -czf <target path>/elastic-backup-MM-DD-YY.tar.gz elasticsearch/
```

Once the tar archive process is complete, you can start the service on this node and move on to the next.

```
service elasticsearch start
```

You can use this check to see the process java is running again:

```
ps -ef | grep java
```

Make sure to repeat this process on all Elasticsearch nodes, and similarly make certain the service returns to operation as expected.

At this point, you can copy the archive files to a safe location however you should make sure your file name convention is distinct and allows for the host where the backup was taken to be identified.

Restore

Start by making sure the archive files for each node is present on each node and copied to `/var/lib`

Log in as the user vbrick on the primary MongoDB node and escalate your privileges using `sudo su -`

Use `service elasticsearch stop` to shutdown the Elasticsearch service on *all* the Elasticsearch nodes in your cluster.

Move your existing Elasticsearch directory to a safe location.

```
cd /var/lib
mv elasticsearch elasticsearch.old
```

Extract the archive/backup file, updating the file name as appropriate for the one you have on hand.

```
tar -xzvf elasticsearch-backup-MM-DD-YY.tar.gz
```

The extraction process should preserve the file ownership but you can confirm/reinforce this with the following command:

```
chown -R elasticsearch:elasticsearch ./elasticsearch
```

Repeat the above steps on the each Elasticsearch node until all data is restored. Once the restoration is complete you can restart the Elasticsearch service on all nodes using `service elasticsearch start`.

MongoDB

The MongoDB tools that are installed when the system was created include a very useful utility named `mongodump` which makes for very easy backup and restoration of a MongoDB database. The `mongodump` utility is the standard process as suggested by the authors of

MongoDB and when possible should be favored over other mechanisms, included any documented in previous Vbrick On-Premises releases. The `mongodump` utility has several advantages, but perhaps the primary benefit is that backups can be taken while the MongoDB service is live and in production use.

The `mongodump` utility is simple to use but isn't ideal on very large databases. For most On Premises installations this tool will be more than adequate.

Note: MongoDB authentication was enabled in version 7.25. The `mongodump` command requires additional parameters to backup most databases and collections. The commands below have been updated to reflect this change. In this and other examples, password input is interactive (there is no actual password specified in the clear with the `--password` option) to ensure best security practices. Always use the credentials that match the instance you are working on.

Backup

Log in as the user `vbrick` or a user with `sudo` or `su` privileges to the primary MongoDB node and escalate your privileges using `sudo su -`

The `mongodump` utility will drop its payload in your current working directory unless otherwise specified so it is easiest to make sure your current working directory is on a partition that has suitable space. Execute the following command:

```
mongodump --oplog --user admin --password --authenticationDatabase admin
```

Archive the dump files into a single file:

```
tar -czf <target path>/mongodb-mongodump-MM-DD-YY.tar.gz dump/  
rm -rf dump/
```

Optionally, copy the backup dump to a safe or off-site location.

Restore

The companion utility to `mongodump` is `mongorestore` and like its cousin requires the `mongod` service be running on the server you're recovering data in. While the restore process can write over existing data, it's better for data integrity purposes to start with an empty database and allow `mongorestore` to rebuild all the collections. Start by making sure the archive files for each node is present on each node and copied to `/var/lib`

Begin by logging into *all* MongoDB nodes, escalate your privileges using `sudo su -` and then shutting down the service with `service mongod stop`.

Starting with the Primary MongoDB node, make a backup of your existing data (if you need to), then create a new data directory, and start the service. MongoDB will rebuild everything it needs as long as its `dataPath` exists.

```
cd /var/lib  
service mongod stop  
mv mongodb mongodb.old  
mkdir mongodb  
chown mongodb:mongodb mongodb  
service mongod start && tail -f /var/log/mongodb/mongodb.log
```

Use **ctrl+c** to exit the tail once MongoDB is fully initialized. At this point, the database is empty so you will need to extract your backup archive in a suitable location, and then run `mongorestore`. For our example we've copied the backup file to `/opt/backup`

```
cd /opt/backup  
tar -xzf mongodb-mongodump-MM-DD-YY.tar.gz
```

```
mongorestore ./dump
```

Once the process completes, you can move on to the next server and repeat the process.

Note: Because this example restore works on an empty database, there is as yet no authentication established so none is required for mongorestore to process the request. Once the restoration is complete, any credentials established previously will be enforced and subsequent commands that require authentication will fail if no credentials are supplied.

Appendix D: Load Balancer Information

Last Updated: November 14, 2018

This document describes how a load balancer should be configured in a Vbrick Rev installation.

It is important that you review the [Installation Prerequisites](#) document which covers machine sizing, setting up DNS, and other important information before beginning or completing the installation.

Load Balancer Concepts

The basic requirements of a Vbrick Rev installation are as follows:

1. Vbrick Rev consists of web servers that reside behind a load balancer VIP listening on HTTP or HTTPS.
 - For HTTP, no special configuration is required.
 - For HTTPS, the load balancer must be the HTTPS endpoint. Plain text HTTP must be used to reach the Vbrick Rev servers.
 - For HTTPS, Vbrick Rev must be configured to **Generate HTTPS Links** during the installation process.
 - Web Socket support is required.
 - X-Forwarded-For header support required
 - No sticky sessions are required.
2. MongoDB and Elasticsearch handle clustering on their own and do not use a load balancer.

Example: HAProxy

HAProxy is a free, open-source piece of software that is suitable to be used as a load balancer.

The Vbrick-provided ISO file provides a fully-configured HAProxy installation.

1. Install HAProxy on a node that can access both Vbrick Rev servers and can be reached by your users.
2. The Vbrick-provided ISO file will edit the configuration file at `/etc/haproxy/haproxy.config`
3. HAProxy will be configured to point its Rev endpoint to Rev Node 1, Rev Node 2, and so on.
4. See the configuration below

This HAProxy configuration file is the same as the one included with our On-Premises Installer. Where you see host names in all upper case (for instance `_REVHOSTFQDN1_`), these are substituted with environment-specific values at the time of installation so you will need to update these to match your own environment before this configuration will work for your instance. The values can either be fully-qualified domain names or IP addresses. Isolated host names will also work as long as they can be resolved by the HAProxy machine.

This file should give you enough structure and example to use to configure your own instance of the service.

haproxy.cfg

```
#
# /etc/haproxy/haproxy.cfg
#
#       VBrick HAProxy Configuration.
#
global
    log          /dev/log local0 notice
#   log          /dev/log local1 notice
    nbproc       4
    maxconn      8000
#   chroot       /var/lib/haproxy
    user         haproxy
    group        haproxy
    spread-checks 5
    daemon
    quiet
    stats        bind-process 1
defaults
    log          global
    option       dontlognull
#   option       dontlog-normal
    option       redispatch
    option       allbackups
    maxconn     256000
    option       httplog
    timeout     connect 5000
    # Long timeout for WebSocket connections.
    timeout     tunnel 1h
#
# Rev HTTP backend.
#
backend rev_backend
    mode        http
    balance     roundrobin
    option      forwardfor
    option      http-server-close
    option      httplog
    #redirect scheme https if !{ ssl_fc }
    server _REVHOSTFQDN1_ _REVHOSTFQDN1_:80 weight 1 maxconn 1024 check
    server _REVHOSTFQDN2_ _REVHOSTFQDN2_:80 weight 1 maxconn 1024 check
#
# Rev HTTP frontend.
#
frontend rev
    bind        0.0.0.0:80
    mode        http
    option      contstats
```

```

    option      forwardfor
    option      http-server-close
    default_backend rev_backend
    #appsession vbrickAccessToken len 105 timeout 1d
#
# Rev HTTPS frontend for HTTP backend.
#
frontend rev_ssl
    bind          0.0.0.0:443 ssl crt /etc/haproxy/cert.pem no-sslsv3 no-tlsv10
    ciphers       ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:ECDH+3DES:DH+3DES:RSA+AESGCM:RSA+AES:RSA+3DES:!aNULL:!MD5:!DSS:!3DES:!RC4
    mode          http
    option        contstats
    option        forwardfor
    reqadd        X-Forwarded-Proto:\ https
    option        http-server-close
    default_backend rev_backend
    #appsession vbrickAccessToken len 105 timeout 1d
    bind-process 2 3 4
# Begin rev_licensing lines.
#
# rev_licensing HTTP backend.
#
backend rev_licensing_backend
    mode          http
    balance       roundrobin
    option        forwardfor
    option        http-server-close
    option        httplog
    #redirect scheme https if !{ ssl_fc }
    server _REVHOSTFQDN1_ _REVHOSTFQDN1_:5555 weight 1 maxconn 1024 check
    server _REVHOSTFQDN2_ _REVHOSTFQDN2_:5555 weight 1 maxconn 1024 check
#
# rev_licensing HTTP frontend.
#
frontend rev_licensing
    bind          0.0.0.0:5555
    mode          http
    option        contstats
    option        forwardfor
    option        http-server-close
    default_backend rev_licensing_backend
    #appsession vbrickAccessToken len 105 timeout 1d
# End rev_licensing lines.

# Begin rev_analytics lines.
#
# rev_analytics HTTP backend.
#
backend rev_analytics_backend
    mode          http
    balance       roundrobin
    option        forwardfor
    option        http-server-close
    option        httplog
    #redirect scheme https if !{ ssl_fc }
    server _REVHOSTFQDN1_ _REVHOSTFQDN1_:2222 weight 1 maxconn 1024 check
    server _REVHOSTFQDN2_ _REVHOSTFQDN2_:2222 weight 1 maxconn 1024 check
#

```

```

# rev_analytics HTTP frontend.
#
frontend rev_analytics
    bind          0.0.0.0:2222
    mode          http
    option        contstats
    option        forwardfor
    option        http-server-close
    default_backend rev_analytics_backend
    #appsession vbrickAccessToken len 105 timeout 1d
# End rev_analytics lines.

#
# HAProxy Statistics interface.
#
listen stats
    bind          0.0.0.0:8081 ssl crt /etc/haproxy/cert.pem no-ssl3 no-tlsv10
    ciphers       ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:ECDH+3DES:DH+3DES:RSA+AESGCM:RSA+AES:RSA+3DES:!aNULL:!MD5:!DSS:!3DES:!RC4
    mode          http
    stats         enable
    stats         hide-version
    stats         realm HAProxy\ Statistics
    stats         uri /
    stats         auth vbrick:VBrickHAProxy1

#
# End of vbrick-haproxy.cfg.
#

```

Appendix E: SSL Certificates

Last Updated: May 29, 2020

SSL certificates are relatively simple to work with and can be generated locally for testing purposes. As a starting off point, all SSL transactions require some form of SSL certificate. SSL certificates can originate from many sources however it should be made clear that only the simplest of certificates is required for the sake of testing SSL or to confirm that SSL works at all.

Private Keys

All SSL starts with a private key, and regardless of the type of certificate, you'll need to generate one.

```
openssl genrsa -out private.key 2048
```

The command above generates a RSA private key named `private.key` with a key-length of 2048 bits. Both the file name and key-length can be adjusted to suit your needs, but you should not have a private key shorter than 1048 bits. Make a note of the path to this file and keep it handy.

Private Key Length

Use the following command to determine your key length. The input in this case uses the HAProxy `/etc/haproxy/cert.pem` file but could be any accepted form of OpenSSL keyed file.


```
sudo openssl rsa -in /etc/haproxy/cert.pem -text -noout | grep Private-Key
```

The number of bits in the key will be displayed in the output.

Certificates and Certificate Signing Requests

SSL certificates are generated from a Certificate Signing Request or CSR which outlines the nature of the certificate you'll need: the company that owns it, where they're located, and what the hostname tied to the certificate will be. All attributes you might consider an aspect of the certificate are actually generated and proposed in the CSR. The CSR is signed with the private key created in the previous step. Once the request is produced, it's on to next choices.

```
openssl req -out mycert.csr -key private.key -new
```

This example creates a CSR file named mycert.csr using the private key we created in the previous step.

When executed in this fashion, the process is interactive and it's the best way to generate the CSR unless you need to do it in bulk (which is beyond the scope of this document). Below is an example of the interactive process.

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:Virginia
Locality Name (eg, city) []:Herndon
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Vbrick Systems, Inc.
Organizational Unit Name (eg, section) []:on-premises
Common Name (e.g. server FQDN or YOUR name) []:qaha.lab.vb.loc
Email Address []:devops@vbrick.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

Operational requirements should drive the need for honest answers to these questions. The only time SSL certificate information becomes critical is when those certificates are being used in public-facing SSL (more on this detail in the next section).

Caution: For the purposes of use with HAProxy, *do not* add a challenge password to your CSR. If you specify something other than an empty value, that value will be required at each initial use of the certificate that is issued/generated from the CSR. This will in turn complicate the HAProxy startup enough that the service may become unreliable. If you've already specified a value, start over; it is far easier to remedy any potential issues in this fashion.

Certificate Generation

To sign or not to sign, that is the question. In fact, someone must sign the CSR or there will be no certificate to generate. If your SSL will be public-facing, it is in your best interest to pursue a certificate signed by one of the leading certificate agencies. Web browsers will always warn about SSL that is suspect: certificates signed by an agency (or person) not in its catalog of trusted sources. SSL will still work under these circumstances, but warning messages do

not exude trust to end users. As a result, self-signed certificates are best left for testing and internal use where excuses are much less likely to generate a support issue.

Generating a self-signed certificate is relatively easy. We've created the first two components we need: the private key and CSR. Now we'll use those two items to sign and produce the certificate.

```
openssl x509 -in mycert.csr -out mycert.crt -req -signkey private.key -days 365
```

Days can be any number you want; with self signed certificates this is not especially critical. We should have now have three PEM-formatted files (the suffixes are arbitrary but somewhat standard for easy housekeeping):

- private.key
- mycert.csr
- mycert.crt

PEM Files

PEM is the acronym for Privacy Enhanced Mail and is a Base64 encoded DER certificate. HAProxy makes use of PEM files. For the purposes of this document any human-readable file produced by the openssl commands on this page, produce PEM files. Up until now we've produced three separate files, and once we have the certificate in hand we only need two; the CSR has served its purpose and can be deleted. Depending on the configuration of your load balancer, you may be asked for the key and the SSL certificate as separate files or as only one. HAProxy uses this latter form of PEM-use (there is no hard rule on this beyond what is required by your load balancer). To create a PEM file for use in HAProxy, simply perform the following commands:

```
mv /etc/haproxy/cert.pem /etc/haproxy/cert.pem.old
cat mycert.crt > /etc/haproxy/cert.pem
cat private.key >> /etc/haproxy/cert.pem
chown root:root /etc/haproxy/cert.pem
chmod 600 /etc/haproxy/cert.pem
```

File permissions haven't been mentioned up until now but they are very critical. Most processes that consume certificates will exit if the security is too lax on files that contain private keys for encryption. Most of the time the file must be owned by root and be unreadable by anyone else. Consult the documentation for your specific load balancer if you have to manipulate files by hand. The instructions above are appropriate for HAProxy.

Restart HAProxy for the new certificate to enabled.

Extras

Strip password from Private Key

```
openssl rsa -in private.key -out new-private.key
```

Test a Certificate File

```
openssl x509 -in mycert.crt -text -noout
```

Debugging a SSL Connection

```
openssl s_client -connect www.mytesthost.com:443
/* Important that you use the full hostname as part of this test */
```

Convert a PEM key and Certificate to PKCS#12 (.pfx .p12)

```
openssl pkcs12 -export -out certificate.pfx -inkey private.key -in mycert.crt
/* Usually this format is required for Microsoft IIS and related solutions.
*/
```

Appendix F: MongoDB Supplemental Information

Last Updated: May 28, 2020

Starting in On-Premises release 7.25, authentication in MongoDB is turned on. This document details how that's deployed in new installations and upgrades as well as additional methods for managing MongoDB when authentication is enforced.

Authentication

There has always been an administrative user provisioned as part of the On-Premises installation process. The values of this attribute had always been intended as a vestigial setting so there is already a *root* MongoDB user with the following details:

- Default MongoDB user: **admin**
- Default Admin password: **admin**

Caution: If you want to change the defaults values for user, password or both, do so *before* proceeding with any upgrade.

MongoDB Client

Working with the MongoDB client when authentication is enabled is relatively simple. Username and password are specified as additional arguments as part of the entered command line.

```
mongo --username admin --password --authenticationDatabase admin
```

This command will interactively prompt for the password for the user specified. The option `--authenticationDatabase admin` must be specified.

Note: The `--password` flag *can* take your password as an argument however, we suggest never entering passwords in this way. Non-interactive passwords may leave clear-text credentials in the shell history.

MongoDB Replication

When MongoDB authentication is enabled, all transactions fall under this umbrella and replication becomes authenticated as a result. Vbrick On-Premises uses a simple private key, specified in `/etc/mongod.conf` that is used by all the servers in the MongoDB cluster to accomplish replication authentication. If you choose to alter the key files created by the Vbrick Setup process, make sure the contents of the key file are identical on all the servers in the cluster. Vbrick Setup is not concerned with the contents of these files, only that they exist.

Authentication key files do not get created in single node installations because MongoDB replication is not in use.

Red Hat 8.1 Dependencies

Red Hat 8 (RHEL8) includes a new Linux kernel version and as a result several of the application binaries we include in our packaging differ between RHEL7 and RHEL8. MongoDB 4.2 support on RHEL8 requires the installation of the python2 package which is available via licensed access to the Red Hat repositories. A screen capture of the package and its dependencies is below.

```
[root@qa-mongo-12a ~]# dnf install python2
Updating Subscription Management repositories.
Last metadata expiration check: 0:43:38 ago on Thu 07 May 2020 07:03:55 PM UTC.
Dependencies resolved.
=====
Package                Architecture Version                                Repository                            Size
=====
Installing:
python2                x86_64    2.7.17-1.module+el8.2.0+4561+f4e0d66a  rhel-8-for-x86_64-appstream-rpms    108 k
Installing dependencies:
python2-libs           x86_64    2.7.17-1.module+el8.2.0+4561+f4e0d66a  rhel-8-for-x86_64-appstream-rpms    6.0 M
python2-pip-wheel     noarch    9.0.3-16.module+el8.2.0+5478+b505947e   rhel-8-for-x86_64-appstream-rpms    1.2 M
python2-setuptools-wheel noarch    39.0.1-11.module+el8.1.0+3446+c3d52da3  rhel-8-for-x86_64-appstream-rpms    289 k
Installing weak dependencies:
python2-pip            noarch    9.0.3-16.module+el8.2.0+5478+b505947e   rhel-8-for-x86_64-appstream-rpms    1.9 M
python2-setuptools     noarch    39.0.1-11.module+el8.1.0+3446+c3d52da3  rhel-8-for-x86_64-appstream-rpms    643 k
Enabling module streams:
python27               2.7
=====
Transaction Summary
=====
Install 6 Packages

Total download size: 10 M
Installed size: 38 M
Is this ok [y/N]: y
```

Extra Commands

Add an Administrative User

Gain access to the Mongo Shell using an existing root user. Once you have a prompt in the shell, you can add a new user:

```
use admin
db.createUser(
  {
    user: "myAdminUser",
    pwd: "a:complicated:password",
    roles: [ { role: "userAdminAnyDatabase", db: "admin" },
"readWriteAnyDatabase" ]
  }
)
```

The user is created and available immediately.

Change an Administrative User's Password

Gain access to the Mongo Shell using an existing root user. Once you have a prompt in the shell, you can alter and admin user's password:

```
use admin
db.ChangeUserPassword("myAdminUser", "my:new:complicated:password")
```

Rev Audit Database User

There may be occasions when you need to create the Rev Audit database user by hand. Once the user is created, Rev will be able to provision the structures it needs with the access as provided.

Start by locating your **PRIMARY** MongoDB server, this command can only be executed there. Using the MongoDB client, gain authenticated access to the cli and execute this command:

```
if (db.getSiblingDB('admin').system.users.find({user:'revUser', db:
'rev_audit'}).count() != 1 ) {
  db.getSiblingDB('rev_audit').createUser(
  {
    user: 'revUser',
    pwd: 'revUser',
    roles: [
      {
        role: 'readWrite',
        db: 'rev_audit'
      },
      {
        role: 'dbAdmin',
        db: 'rev_audit'
      }
    ]
  }
);
}
```

Appendix G: Operating System Upgrades

Last Updated: June 5, 2020

This appendix covers the upgrade considerations for each of the platforms Vbrick supports.

Windows Server

For Windows Server update from 2012 to 2016 or 2016 to 2019 please follow Microsoft Windows Server Update documentation and process.

- [Windows Server 2012 → 2016](#)
- [Windows Server 2016 → 2019](#)

If you are upgrading Windows after you have upgraded Rev to 7.34, then please carry out the following steps *after* upgrading Windows and *prior* to starting the Rev Services.

1. Open a Windows Explorer window to `C:\Rev\VbrickPlatform.Runtime.Host`
2. Choose one of the following scenarios:
 - For **Multi-Node** installations, right-click the file named `RabbitMQCluster.bat` and select “Run as administrator.”
 - For **Single-Node** installations, right-click the file named `SingleNode.bat` and select “Run as administrator.”

For Windows Server 2012 → 2016:

- Rename the farmhash folder (`C:\Rev\Rev-Analytics\node_modules\farmhash`) to farmhashOLD for example and then rename the farmhash-w2016 folder to farmhash.
- Restart runtimes and the Analytics if already running.

For Windows Server 2016 → 2019:

- Rename the farmhash folder (`C:\Rev\Rev-Analytics\node_modules\farmhash`) to farmhashOLD for example and then rename the farmhash-w2019 folder to farmhash.
- Restart runtimes and the Analytics if already running.

Linux Servers

Both Canonical and Red Hat offer a pathway to their next major versions. For the versions Vbrick supports, please follow the vendor documentation:

- [Red Hat 7 to Red Hat 8](#)
- [Ubuntu 16.04 to 18.04](#)

Caution: Both Red Hat 8 and Ubuntu 18.04 use the v4 Linux kernel. As a result of this many packages Vbrick includes in On Premises releases now have different binaries that support these newer OS versions.

The best way to process an OS upgrade is to first upgrade the systems to Vbrick Rev 7.34 On Premises, then go through the OS upgrade process. Once complete you should take a full inventory of the systems to make sure everything is intact.

Ubuntu 16.04 → 18.04:

In testing, the upgrade removed Elasticsearch from all the servers where it was present. The Elasticsearch data is untouched but you must make sure to keep the old (existing) configurations when reinstalling Elasticsearch.

Red Hat 7 → Red Hat 8:

In testing, the HAProxy node (the one used for the 7.34 upgrade and had NodeJS installed) could not be upgraded without removing NodeJS before the upgrade. NodeJS can be safely removed once the 7.34 upgrade is complete, there is no need for this package during production use.

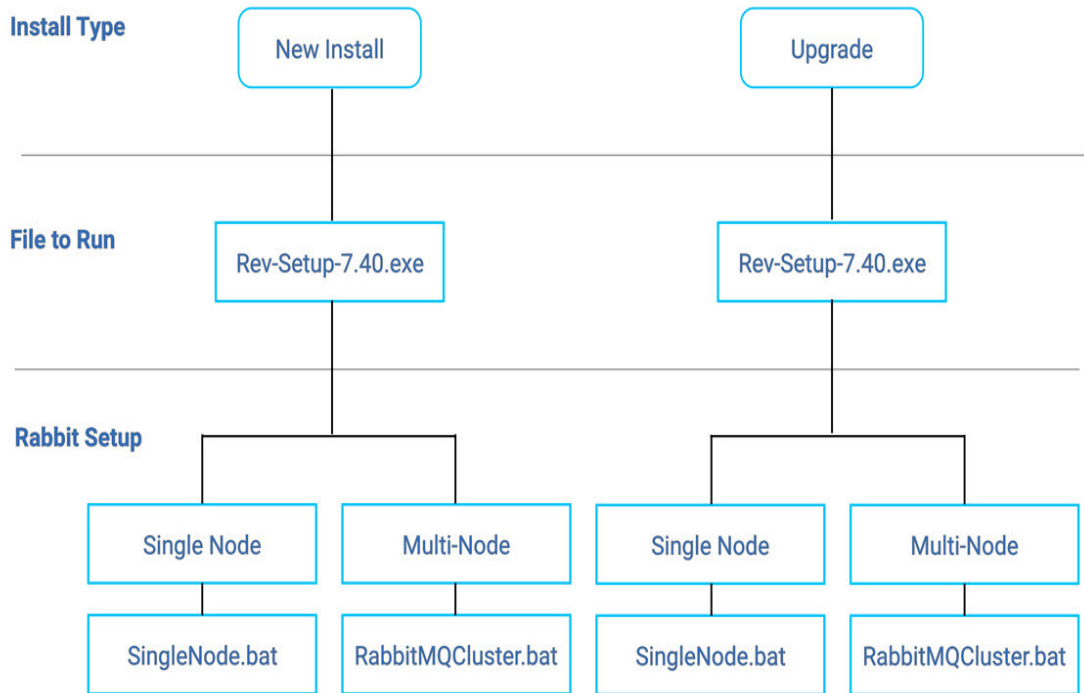
Appendix H: Installation Steps Quick Reference

Last Updated: June 10, 2021

This installation flow chart is intended to provide a quick reference on the steps and commands necessary to install On-Premises Rev in both Windows and Linux.

Be advised this is a quick reference only. You should *always* read the installation guide fully to make sure you understand the steps, procedures, and process completely before attempting a new installation or upgrade.

Windows Quick Reference Installation Guide



Linux Quick Reference Installation Guide

