MicroStrategy | 2021

System Administrator

Planning and Optimizing the Analytics Infrastructure

CONTENTS

1

The Intelligent Enterprise

Your company, InfiniRec, is a fitness club technology startup that has decided to transform into an Intelligent Enterprise. As MicroStrategy users, InfiniRec wants to leverage its existing investments and successfully deliver powerful analytics and mobility solutions across the enterprise.

The CEO of InfiniRec has selected you to take on the System Administrator role, which is tasked with the development of guidelines related to infrastructure procurement, maintenance, monitoring, and optimization.

In this chapter, we will review:

- **The Intelligent Enterprise**: a data-driven organization that maintains standards to drive the implementation of analytics solutions.
- **Your role as the System Administrator**: create standards that guide infrastructure administrators in establishing and maintaining the infrastructure that houses the analytics solution.

Introducing the Intelligent Enterprise

The Intelligent Enterprise is a data-driven organization that designs and implements standards to create and maintain effective analytics solutions while

promoting data consumption across the enterprise. This fosters growth and development, with a focus on data governance and alignment of strategic business goals with technology investments.

Transforming into an Intelligent Enterprise requires the right tools and structure to balance traditionally counteractive forces—agility and governance, convenience and security, ease of use and enterprise functionality—all critical capabilities that the MicroStrategy platform is positioned to support with its unique intelligence architecture.

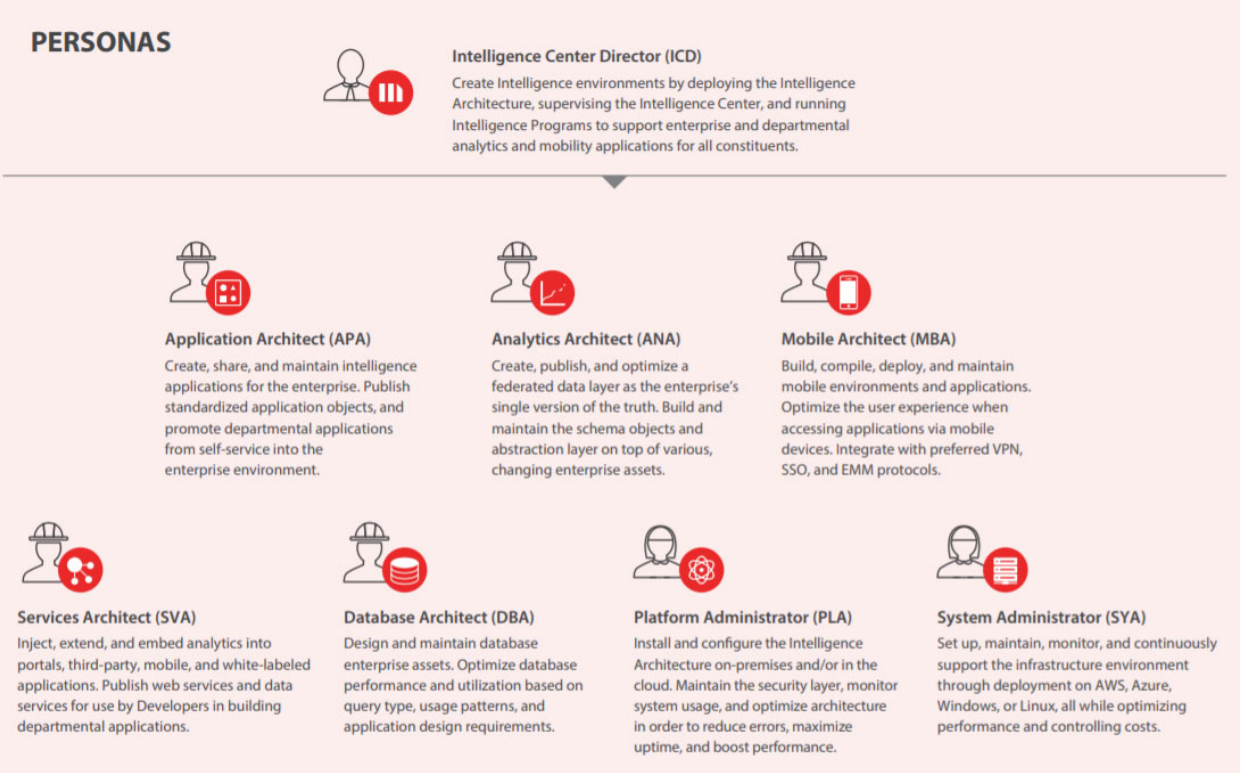
A successful Intelligent Enterprise creates corporate standards to:

- Drive the adoption and success of enterprise Business Intelligence (BI).
- Coordinate BI implementations.
- Maintain sound data governance and a single version of the truth.
- Provide a formal approach to documenting processes, creating content, and ongoing maintenance.
- Ensure that BI is aligned with enterprise strategy.

Along with quick and easy ad-hoc departmental solutions, MicroStrategy has the robust, proven ability to support high-scale deployments and establish a single source of the truth. MicroStrategy's tools include data-governance features, administrative controls, and management capabilities with the enterprise platform software, all critical to the Intelligent Enterprise.

The Intelligence Center

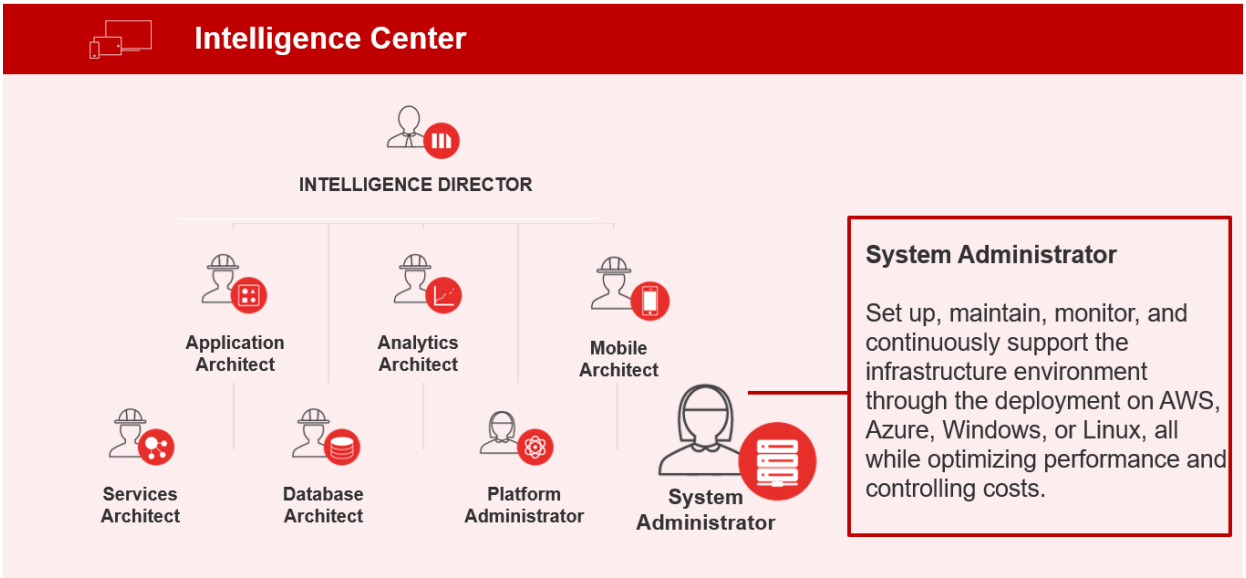
At the core of the Intelligent Enterprise, the Intelligence Center is made up of a team of expert architects who define standards, develop corporate objectives, and provide guidance across the enterprise.



With the collective knowledge to maximize analytics investments, the Intelligence Center drives optimization of system architecture, upgrades, configuration, performance, scalability, and stability.

Your Role: System Administrator

As the System Administrator in your Intelligent Enterprise, you are responsible for creating and communicating standards for the administrative tasks related to your computing infrastructure.



The standards that you create must promote your corporate policies and values. The administrative tasks in your organization may include:

- **Provisioning the Intelligent Enterprise infrastructure**— Provisioning, installing, and configuring the following components in the on-premise or cloud environments:
  - hardware
  - operating systems
  - networks
  - file systems
  - applications
  - firewalls
  - web servers
  - monitoring and alerting tools

- **Configuring connectivity**—Establishing connectivity between various components in the environment, including ODBC Connectivity, Authentication Providers, and other required objects.
- **Monitoring environments**—Performing regular system monitoring to verify the integrity and availability of all server resources and key processes. For example, environments must be monitored to verify scheduled jobs such as backups.
- **Maintaining environments**—Applying operating system patches and upgrades, upgrading administrative tools and utilities, configuring new services, upgrading hardware, performance tuning, and optimizing resources.
- **Troubleshooting environments**—Reviewing system and application logs to determine the root cause of issues as they arise.
- **Backup, recovery, and failover plans**—Creating backup and recovery strategies and failover plans to ensure business continuity in case of failures or emergencies.

Your team

Your main objective as the System Administrator in your organization is to develop standards that define clear duties and best practices for the members of your team who perform system administration work. Through these standards, you guide administrators in creating and maintaining the infrastructure that supports a scalable, dependable, and high-performing analytics solution.

The standards that you develop will ensure that consistent practices are implemented by infrastructure administrators on your team to produce reliable results.

Your experience and qualifications

In addition to general administration skills, as the System Administrator, you should have the following experience and qualifications:

- Extensive knowledge of the MicroStrategy security model and platform optimization practices, including:
  - object migrations
  - security filter development
  - Intelligent Cube requirements

    - metadata creation
    - VLDB settings
    - Administrative tools such as Command Manager and Integrity Manager.
- Performance tuning and troubleshooting experience.
- The ability to debug and fix technical issues.
- Deep SQL knowledge.
- The ability to interact with business users for requirement gathering and technical support.

You can gain this knowledge through a combination of experience and training through the following MicroStrategy classes:

- Administration for Enterprise Analytics
- Administration for MicroStrategy on Cloud
- Analytics Performance Tuning
- What's New in MicroStrategy for Administrators
- Advanced Big Data Administration
- System Administrator: Planning and Optimizing the Analytics Infrastructure
- System Administrator (SYA) Certification

Hiring infrastructure administrators

The infrastructure administrators that you hire for your team will follow your standards and guidelines to consistently and predictably set up and maintain the infrastructure required to support the MicroStrategy platform and the software that supports the analytics ecosystem.

To ensure that administrators are able to implement your standards and guidelines, you must hire team members that have a specific set of skills and ensure that they are aware of their roles and responsibilities.

For example, you might require that the administrators on your team have the following skills and experience:

- Hardware and software procurement
- Hardware, software, and network installation and maintenance

- User management
- System management
- Customer service and communication
- Technical troubleshooting
- Monitoring
- Multiple operating systems (Windows, Linux, etc.)
- Cloud provisioning and maintenance (AWS, Azure, etc.)
- Backup and recovery
- MicroStrategy platform experience

Once you assemble a team of qualified infrastructure administrators, you must ensure that they understand their responsibilities and perform their tasks using consistent workflows. To do this, you will train the team using the standards and guidelines you will create throughout this class.

In this class, you will revisit system administration topics framed through the holistic lens of a System Administrator in your new Intelligent Enterprise. The course focuses on key competencies that are required to help you succeed in your new role. These skills will help you support your organization to create and maintain standards in developing the hardware and software infrastructure that serves as the foundation for the organization's analytics applications.

2

Sharing Knowledge with the Enterprise

As InfiniRec transitions to an Intelligent Enterprise, all members of the organization who have a stake in the success of the MicroStrategy platform must be able to communicate freely and share knowledge. Keeping communication channels open ensures that all members of the organization are working toward a common goal, and mitigates the risks associated with conflicting priorities.

In this chapter, we will review:

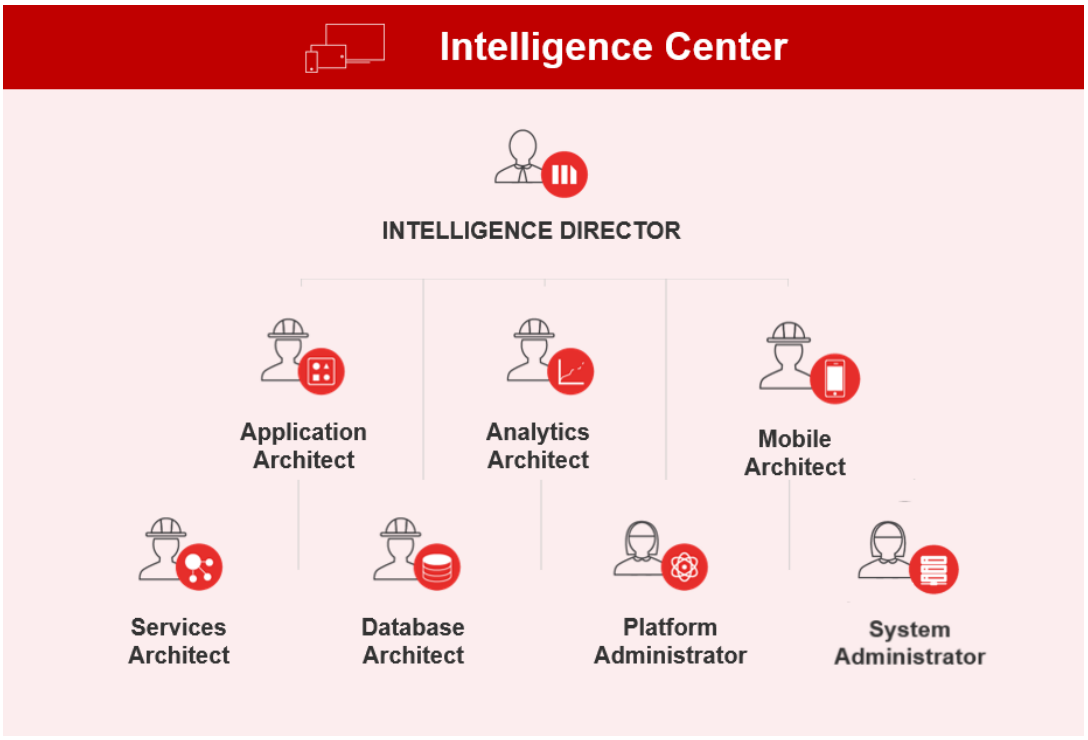
- Coordinating with fellow architects in the Intelligence Center
- Creating a documentation repository to store
  - The standards documentation that you create
  - The infrastructure artifacts for each environment
- Establishing a training schedule to keep your team in tune with standard procedures and infrastructure modifications

Coordinate with fellow architects

Although each role in the Intelligence Center is responsible for a subset of the analytics solution, you must keep the communication lines between yourself and

other architects open to develop and maintain cohesive products. Plan regular meetings with other Intelligence Center architects to communicate proposed changes to the infrastructure, and to review and update the standards you have created for your own team.

Specifically, maintain communication with the following roles:



- **Intelligence Director**: Deploys the Intelligence architecture and supervises the Intelligence Center
- **Application Architect**: Creates, shares, and maintains analytics applications
- **Mobile Architect**: Builds, compiles, deploys, and maintains mobile environments and applications
- **Analytics Architect**: Creates, publishes, and optimizes a federated data layer as the single source of truth
- **Services Architect**: Injects, extends, and embeds analytics into portals, third-party, mobile, and white-labeled applications
- **Database Architect**: Designs and maintains database enterprise assets
- **Platform Administrator**: Installs and configures the MicroStrategy platform

Can you think of any infrastructure administration work that will require coordination with at least one of the above roles?

Supporting Intelligence Center Architects

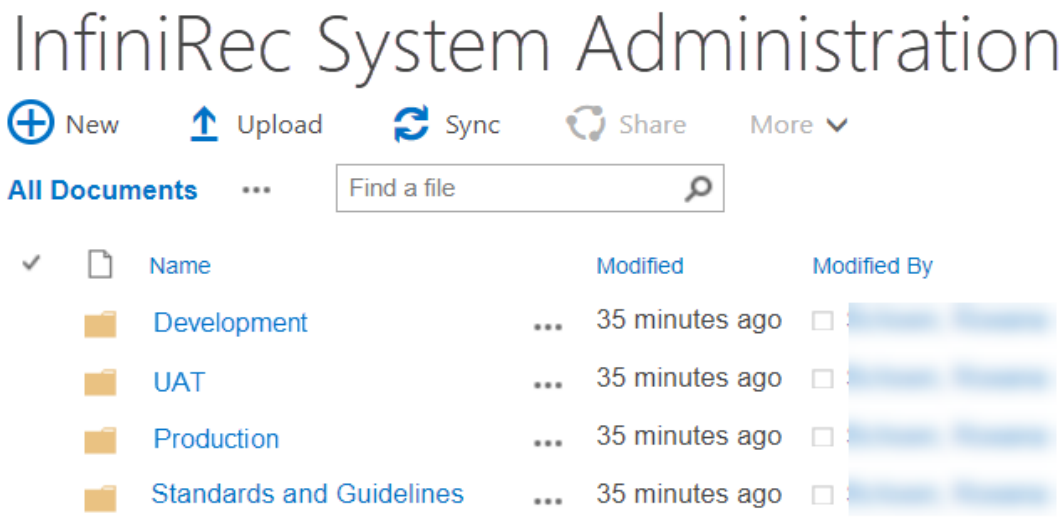
As the System Administrator, you are responsible for supporting the Intelligence Center Architects in building and operating infrastructure environments that adhere to the Intelligence Programs best practices.

As part of your role, you must also ensure that other architects are able to publish and distribute their analytics applications and deliver them to end users. Work with other architects to ensure that the infrastructure environments and networks are available and capable of transmitting, displaying, and storing analytics products.

Create a documentation repository

In addition to the communication channels between yourself and other architects in the Intelligence Center, you must construct a communication channel to share documentation between your team members, other teams, and project stakeholders. To do this, create a central documentation repository using a documentation management platform like SharePoint that enables you to maintain documents, control access, and track changes. You can store and track documentation using your organization's preferred method, such as a wiki, issue tracking software, or file management system.

Providing access to documentation is a vital component of project tracking and continuity. The document repository should contain folders for each infrastructure environment, as well as a dedicated folder for standards and guidelines, as in the following sample:



The documentation repository maintains the documentation outlined in the following sections.

Which documentation repository products does your organization employ?

Which repository features are most important to you?

Intelligent Enterprise standards documentation

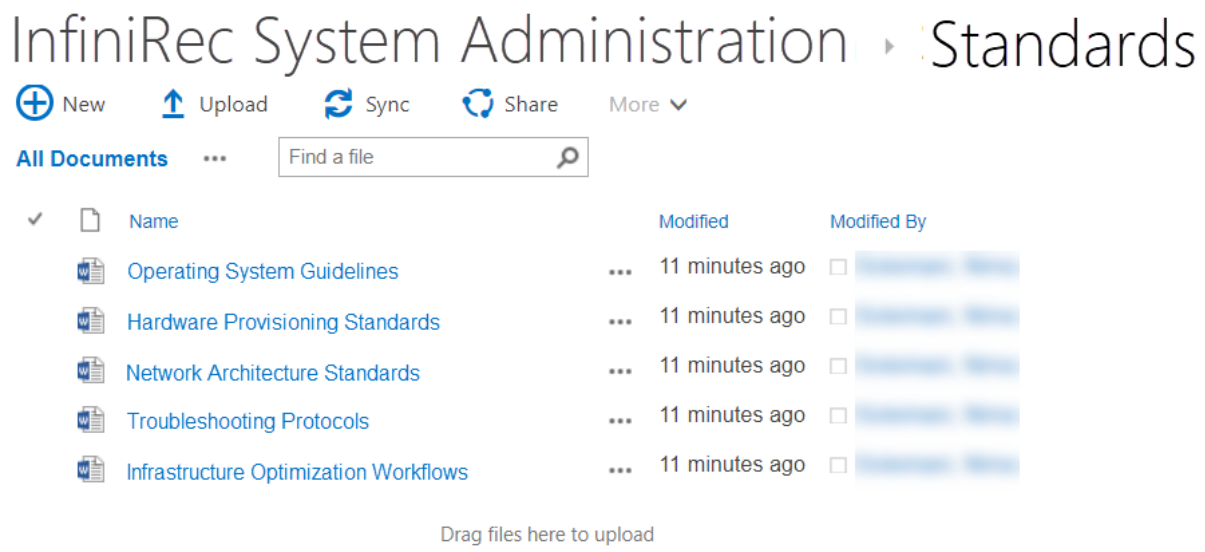
As you develop standards and guidelines for infrastructure administrators, share them with your team, as well as other architects in the Intelligence Center. Distributing your standards ensures that all architects are properly coordinating efforts, and provides you with an opportunity to verify that your team practices align properly with other teams. The documentation that you create also enables administrators on your team to reliably produce consistent infrastructure artifacts.

For example, you might document standards and guidelines in the following example documents, which should guide developers in creating artifacts for each environment:

Standards Document	Description	Per-Environment Artifacts
Operating System Guidelines	• Operating system providers and version for each platform component • Installation and maintenance instructions	• Service Level Agreement (SLA)
Hardware Provisioning Standards	• Sizing guidelines for Intelligence Server, Web Server, Mobile Server, clients, and database server machines • Service account information • Processor, memory, and storage types and capacities • Maintenance instructions	• Service Level Agreement (SLA)
Network Architecture Standards	• Physical location and distance between all servers • Service account information • Secure certificate and port information • Data source connections and maintenance • Architecture maintenance • Web server installation and maintenance	• Topology diagram

Standards Document	Description	Per-Environment Artifacts
Troubleshooting Protocols	• Logging procedures • Monitoring tools, procedures, and alerts • Troubleshooting workflows	• Event logs • Server dump files
Infrastructure Optimization Workflows	• Intelligence Server settings optimizations • Network utilization optimizations • Processor, memory, and other hardware resource additions • Operating system, network, and file system optimizations • Hardware maintenance • Failover plans	• N/A

Maintain standards documentation in its own folder within your documentation repository, as in the following image:



How can you ensure that your standards and guidelines remain relevant as future MicroStrategy updates are released?

Establish a training schedule

Infrastructure requirements evolve over time, as will your administration guidelines. The standards that you document must be delivered to your team through regular training sessions. You must also keep all team members abreast of changes made to the infrastructure environments and components.

To ensure that your team is consistently able to apply administrative practices, create a training schedule and develop courses that convey the guidelines that you have documented. Regularly scheduled training is also a great way to keep your team members updated on recent changes to the infrastructure, as well as development plans for the future.

3

Laying the Foundation and Creating Your Infrastructure

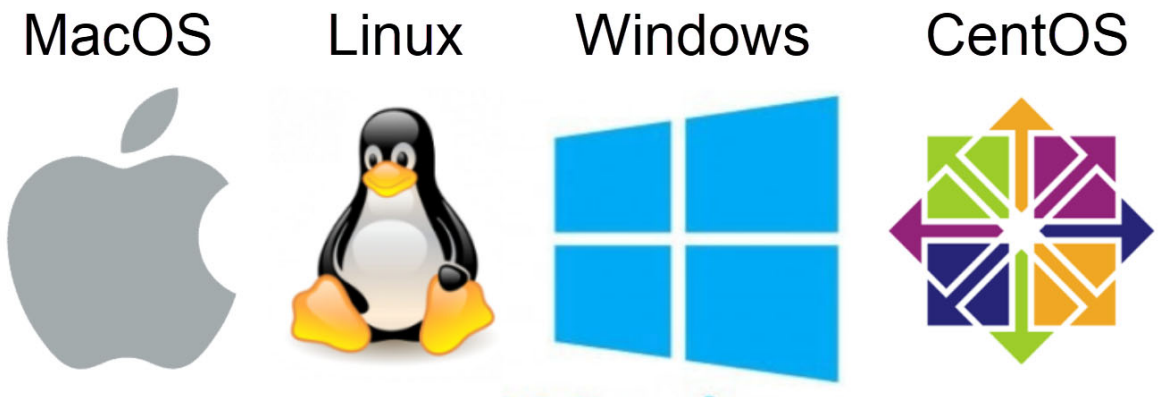
InfiniRec has decided to begin the MicroStrategy platform integration project, and you are ready to plan the infrastructure that is required support the ecosystem. Your goal at this stage is to understand the role of various infrastructure components and develop documentation to help administrators on your team make appropriate infrastructure choices for your environments.

In this chapter, you will learn to create standards that help administrators perform the following infrastructure planning processes:

- **Operating system lifecycle management**: selecting and maintaining the operating system for server and client components.
- **Hardware lifecycle management**: sizing and maintaining machines.
- **Network architecture configuration**: securing and establishing the physical location and connections between infrastructure components.
- **Directory services management**: creating and maintaining a user directory that can be integrated with the MicroStrategy platform.
- **Server file system creation**: establishing the repository required to store MicroStrategy system files.
- **ODBC connection maintenance**: using the appropriate drivers to connect to your repositories.

Identifying the operating systems that run your machines

The operating systems installed in your environments help you manage and maintain your machines and run the MicroStrategy platform's tools and services. You can run MicroStrategy clients and servers on Windows, MacOS, Linux, CentOS, Android, or iOS. Each server or client has its own unique requirements for operating system type and version.



The operating systems that you choose to install in your environments must be compatible with your corporate policy, your hardware resources, and the MicroStrategy platform version that you plan to install. To view the certified operating systems for each MicroStrategy server or client, see the MicroStrategy ReadMe for your platform version.

As you take into account your organization's unique variables and the minimum MicroStrategy requirements, create a corporate policy that conveys operating system installation and maintenance practices to all administrators. For example, you might dictate that Intelligence Server must be installed on a machine that runs Red Hat Enterprise Linux, security patches must be applied every Friday morning, and the operating system must updated quarterly.

The operating system policies that you create for your organization might include the following aspects of operating system installation and maintenance:

- Specify the operating system to install on the machine that runs each MicroStrategy server and client:
    - Specify the type of operating system that administrators can install, such as Windows or Linux.

    - Specify the version of the operating system that administrators can install, such as Windows Server 2016 or Red Hat Enterprise Linux version 7.5.
    - Specify the required filesets, security patches, and libraries.
- Establish a schedule for applying the latest patches and security updates to the operating system
- Create a service account in the operating system and provide the credentials to administrators who will install and register MicroStrategy components.

What is the distribution of operating systems for client machines in your organization?

Have you installed Intelligence Server and other server components on more than one operating system? How has this impacted your experience?

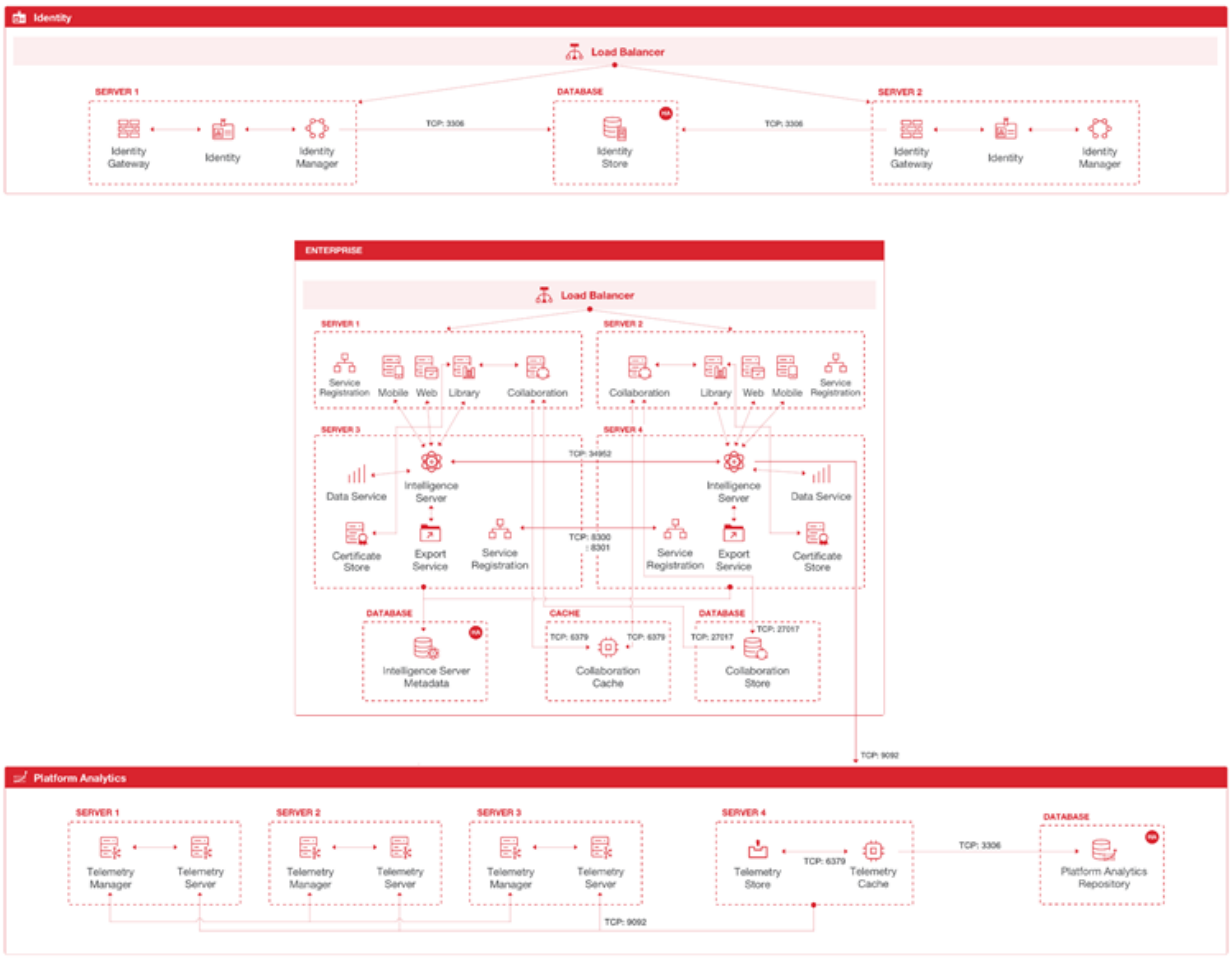
Selecting hardware resources to accommodate your organization's demands

The machines where you install the MicroStrategy platform components require careful planning to adequately operate the software and quickly and reliably deliver data to end users. To ensure that each component of the MicroStrategy platform functions optimally, you must consider several variables when sizing your hardware.

To help administrators in your organization procure and provision adequate hardware for the MicroStrategy platform, develop sizing guidelines for the following components:

- Intelligence Server
- Web Server and Mobile Server
- MicroStrategy clients
- Database servers

You can see the relationships and connections related to these components in the following architecture diagram.



The hardware sizing guidelines that you create for each of these components is based on unique factors such as budget, corporate standards, user concurrency, performance expectations, and so on.

To understand the minimum requirements for each platform component, see the MicroStrategy Installation and Configuration Guide for the version you plan to install: https://www2.microstrategy.com/producthelp/current/InstallConfig/en-us/Content/System_sizing_guidelines.htm

Use the minimum requirements as a starting point in conjunction with the unique needs of your organization and user base to develop your guidelines.

When creating your sizing guidelines, consider variables like CPU speed, CPU type, operating system version, service upgrades, file space, and physical and swap memory. You should also consider your expected environment usage and performance requirements to ensure your system is sized appropriately.

Selecting the appropriate MicroStrategy deployment

The MicroStrategy platform can be deployed on your local infrastructure (on-premise), in the cloud, or in a hybrid configuration.

The deployment option you select depends on your organization's allocation of financial resources, manpower, infrastructure expertise, desired timeline, and ability to house and maintain hardware resources.

If you choose to deploy the platform on-premise, your organization is responsible for all aspects of the environment, including:

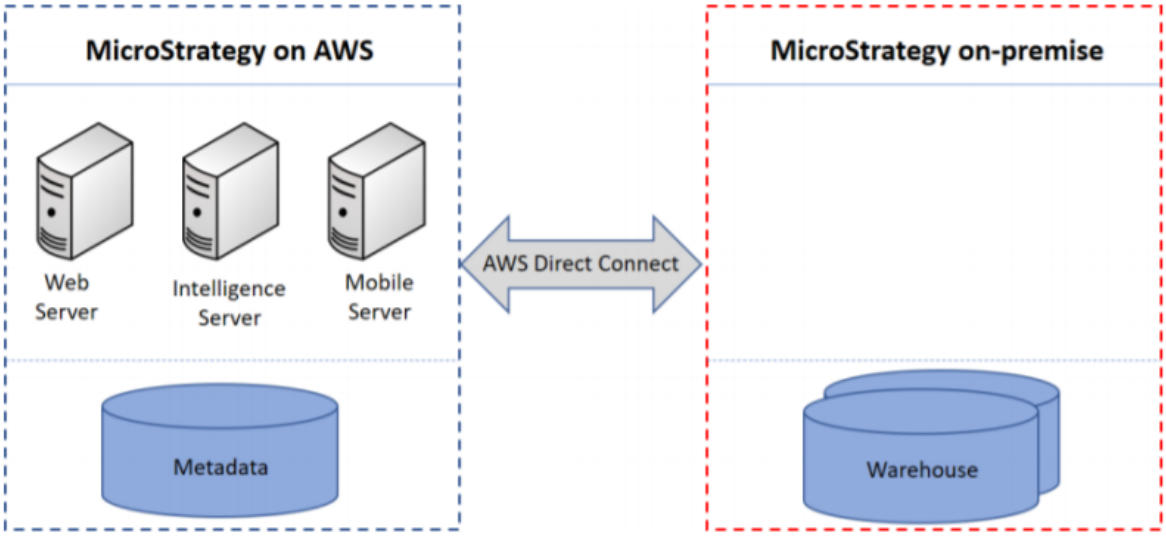
- Maintenance and upgrades
- Security management
- Performance
- Monitoring and troubleshooting

Alternatively, you can deploy the MicroStrategy platform in the cloud. This option enables you to quickly deploy your analytics solution and offload many of the administrative responsibilities. For example, a cloud deployment offers the following conveniences:

- Hardware and software management
- Ability to quickly scale resources to fit user demand
- Minimal initial investment

- Ability to leverage a fully-managed option that encompasses monitoring, troubleshooting, maintenance, and upgrades

A hybrid approach enables you to leverage the MicroStrategy platform in the Cloud while maintaining your data warehouse on-premise. This deployment option enables you to maintain sensitive data on your own infrastructure and avoid the costs associated with moving your data to the cloud.



Include the deployment option as part of your hardware standards documentation. Convey the reasoning behind the deployment strategy, as well as its advantages and disadvantages. This documentation helps your team understand your organization's deployment, and guides decision making for deployment changes in the future.

Has your organization leveraged MicroStrategy Cloud yet? What benefits have you experienced after transitioning to the cloud?

Demo 3.1: Create a MicroStrategy Cloud environment

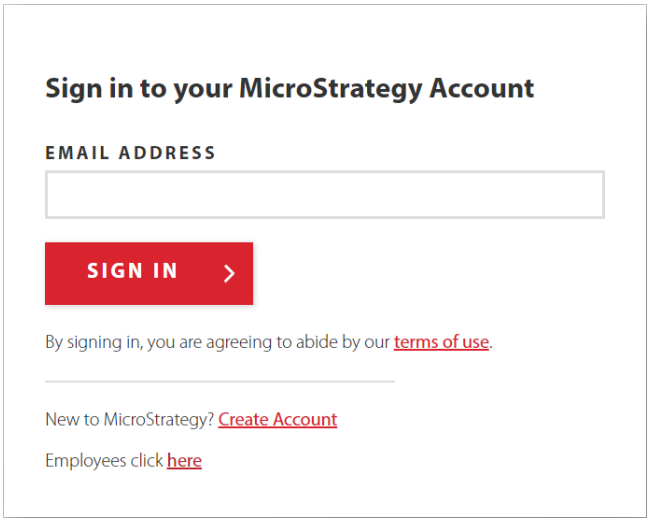
InfiniRec wants to quickly get a test environment up and running to understand how the MicroStrategy platform works. The organization doesn't currently have the resources to dedicate to an on-premise environment.

Deploying MicroStrategy in the cloud enables you to quickly create and deploy the required infrastructure and software installations. Your cloud-based deployment allows you to only pay for the infrastructure resources that you need. You can start up, shut down, and resize your environments to accommodate changes in demand.

In this exercise, follow along as the instructor deploys an environment that includes one Linux-based server and a Windows-based client machine.

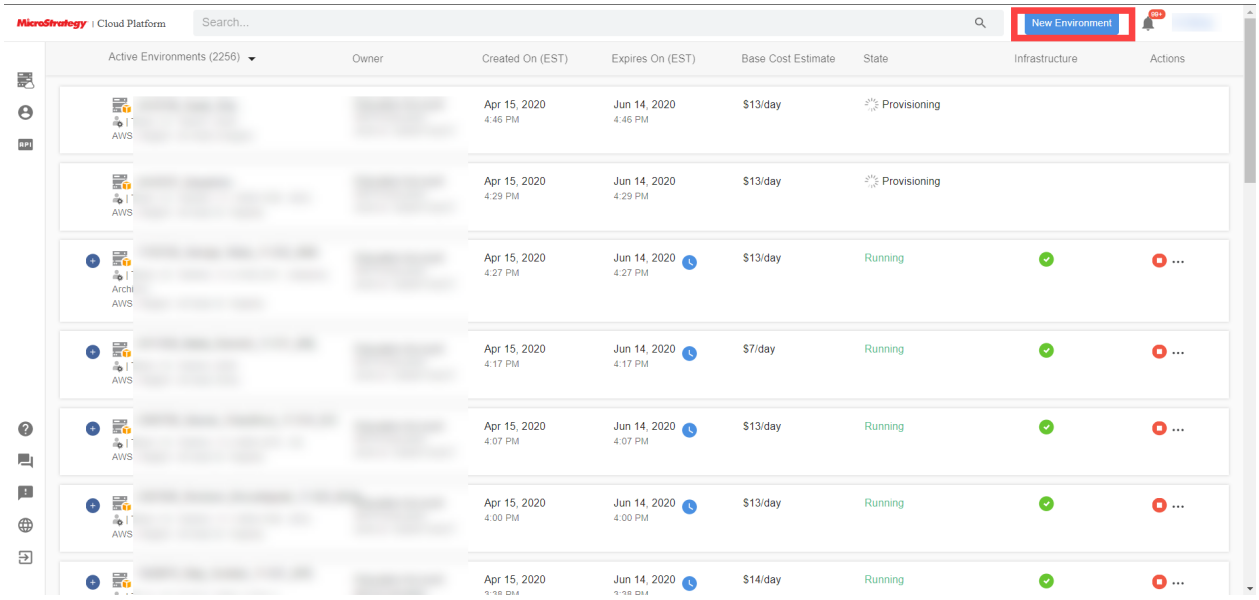
Deploy a MicroStrategy Cloud environment

1 From your browser, open **https://provision.customer.cloud.microstrategy.com**



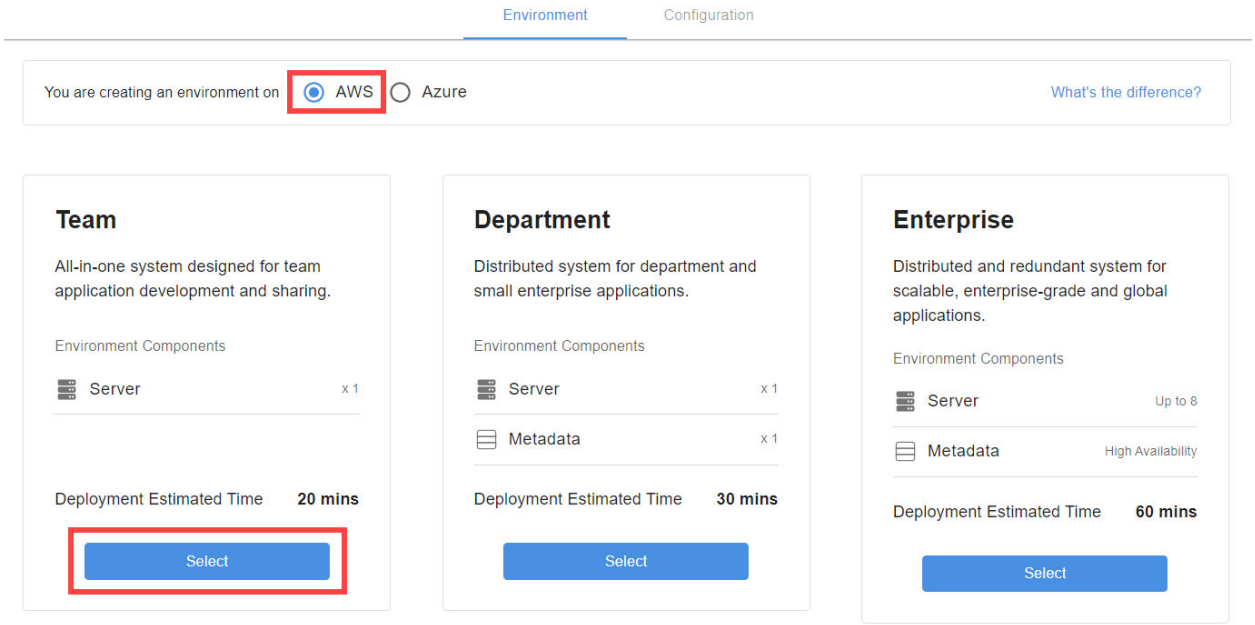
2 Enter your MicroStrategy Cloud Console credentials and click **Sign In**.

3 At the top of the screen, click **New Environment**.

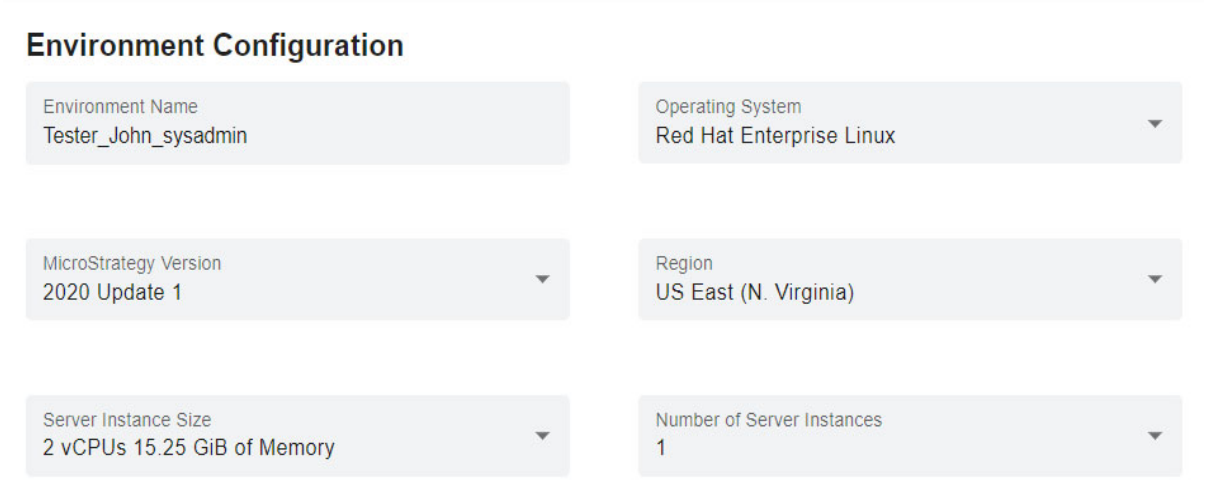


4 To deploy your environment in your Amazon Web Services infrastructure account, click **AWS**.

5 To create an environment suited for a small team, in the Team area, click **Select**.



6 In the Environment Configuration area, do the following:



a In the **Environment Name** box, type ***last name_first name*_sysadmin**. For example, Tester_John_sysadmin.

b From the **Operating System** drop-down list, select **Red Hat Enterprise Linux**.

c From the **MicroStrategy Version** drop-down list, select the latest MicroStrategy release.

d From the **Region** drop-down list, select **US East (N. Virginia)**.

e From the **Server Instance Size** drop-down list, select **2 vCPUs 15.25 GiB of Memory**.

f From the **Number of Server Instances** drop-down list, select **1**.

Because this demo deploys an environment to a corporate MicroStrategy account, an AWS is not required. Before you create an environment in your own organization, create the AWS infrastructure where the platform will be deployed and then specify your AWS account during the environment creation process.

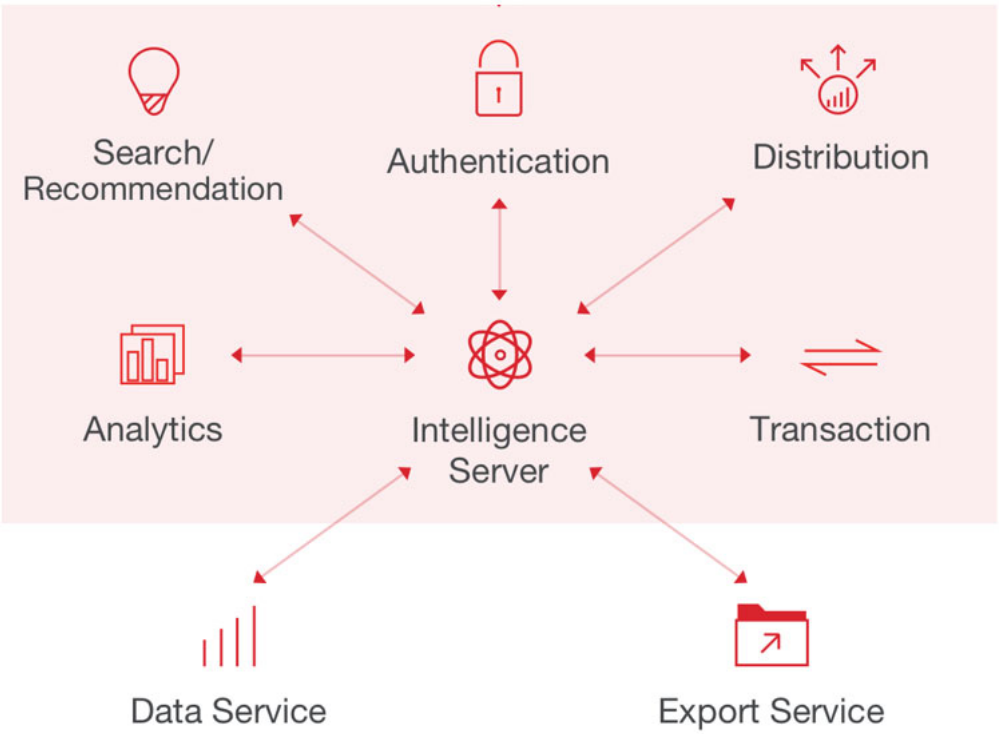
7 Click the **Add MicroStrategy Developer** check box.

The Developer Instance is a second server that includes a Windows installation, administrative applications, and several MicroStrategy platform tools and clients.

8 Click **Create Environment**. The environment will provision in about 20 minutes.

Sizing the Intelligence Server to support demand

The MicroStrategy Intelligence Server performs a wide variety of memory-intensive processes, including caching, scheduling, and cube generation. Because Intelligence Server is the central and most critical component of the MicroStrategy platform, the server where you install it must be carefully provisioned to include an appropriate amount of hardware resources.



Intelligence Server is licensed based on the number of processors (CPU). Thus, Intelligence Server can only be installed on machines with a maximum number CPUs. If you try to install the product on a machine with a larger number of processors than stipulated in your license, installation fails.

As you determine sizing guidelines for the Intelligence Server machine, consider your environment's unique variables. For example, while the minimum RAM requirement for an Intelligence Server machine is 4 GB, 64 GB or more RAM is required to fully leverage performance-improving technologies such as MicroStrategy OLAP Services, and to support optimal reporting performance.

To determine the unique hardware requirements for your organization's Intelligence Server machines, create a test environment and perform system

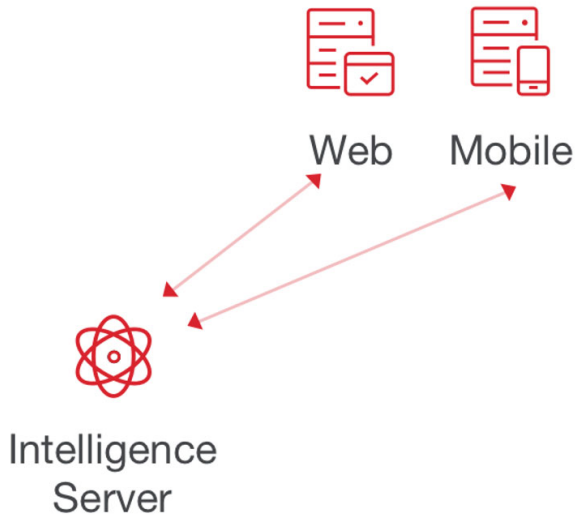
sizing and capacity planning tests. The Intelligence Server hardware provisioning guidelines you create might include the following:

- Specify the processor type, processor speed and the number of processing units (processors or cores) required for the Intelligence Server machine.
- Determine if hyper-threading (for Windows) or multi-threading (for Linux) needs to be enabled to achieve optimal performance.
- Establish physical and swap memory requirements required for memory-intensive operations such as storing data in Intelligent Cubes.
- Convey storage (hard disk space) requirements.

What factors must be weighed when procuring the processor, memory, and disk for Intelligence Server machines?

Sizing the Web Server and Mobile Server to support user concurrency

The Web Server and Mobile Server enable MicroStrategy users to access reports and dossiers through a web browser or mobile application. These servers enable the dispersion of analytics throughout your organization.



To create hardware provisioning guidelines for the Mobile Server and Web Server, identify the minimum requirements in the MicroStrategy Installation and Configuration Guide for the platform version you plan to install. Use the minimum

requirements as a starting point in conjunction with your organization's unique variables to develop your guidelines.

The Web Server and Mobile Server hardware provisioning specifications that you create for your organization might include the following guidelines for each server machine:

- Specify the processor type, processor speed and number of processing units (processors or cores).
- Designate the physical and swap memory requirements.
- Identify the required storage (hard disk space).

Sizing client machines to accommodate all user activity

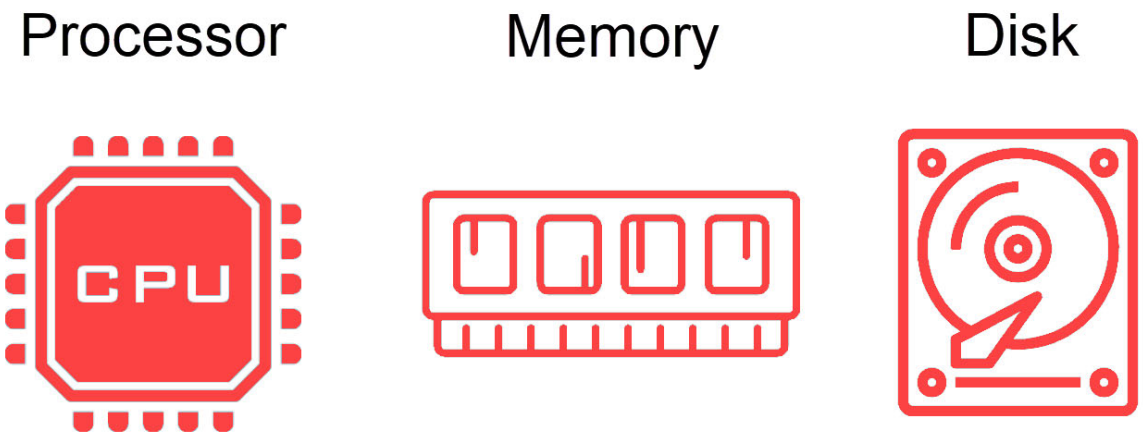
The client machines in your environment are used to access MicroStrategy projects through client tools like Workstation. MicroStrategy installs a set of common files that are shared when installing multiple MicroStrategy products on the same machine. As a result, you should consider the additional storage requirement for the common files that are shared among all client products. The following image shows some examples of MicroStrategy clients.

To ensure the installation process is completed successfully, all MicroStrategy platform and hotfix installations require a certain amount of disk space for the installer itself. This is in addition to any component or common file storage requirements listed in the Installation and Configuration Guide.

Based on your organization's unique practices and needs, develop guidelines to provision machines for MicroStrategy clients like Workstation, Command Manager, Object Manager, and so on. To establish the hardware specifications for your organization, start with the requirements outlined in the Installation and Configuration Guide, and include additional resources to address the demands placed on the machines by your users process. At a minimum, the standards that you create should consider the following components of client machines.

- Processor type and speed.

- Memory type and capacity. The memory requirements you establish should consider the number of concurrent programs that need to run on client machines, and the need to run memory-intensive operations such as data storage in Intelligent Cubes.
- Storage space requirements. Consider the storage requirements for each tool, as well as the common files required to run each application.



If the guidelines that you create recommend installing multiple MicroStrategy products on a single machine, you must consider the disk space requirements for MicroStrategy server and client installations. The standards that you create might include:

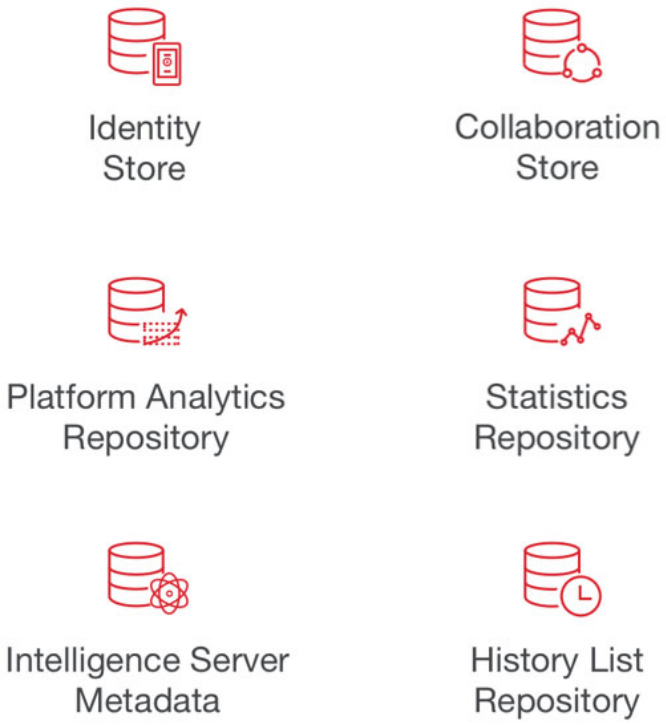
- Storage requirements for installation of MicroStrategy common files
- A storage allotment in a temporary directory for installation files
- Storage space requirements for Cache, History files, and Cube files to support adequate response times for commonly accessed reports, documents, and dossiers.

Which roles in your organization require the greatest allocation of hardware resources?

Sizing database servers to suit data processing and transmission needs

Database servers store the data that is retrieved and eventually displayed in MicroStrategy reports and dossiers. For example, the following image shows

some of the database servers that might exist in your MicroStrategy environments.



To ensure that data is retrieved efficiently and MicroStrategy users are making decisions based on the latest data, create standards for the database server machine hardware. The hardware decisions that you make depend on the amount of data stored, the number of concurrent users, your organization's requirements for data refresh intervals, and so on.

Your database server hardware guidelines might include the following information:

- Work with the Database Architect to identify the required processor type, processor speed, number of processors, memory, and storage (hard disk space) for database server machines.
- Create a service account for the Database Architect to install database server software.

Exercise 3.2: Identifying infrastructure requirements for the MicroStrategy platform

The Intelligence Center architects have experimented with the cloud-based environment, and they now want to explore the infrastructure requirements for an on-premise deployment.

When you install or upgrade the MicroStrategy platform, you must understand the minimum requirements for the machines where platform components will be installed. Your infrastructure planning guidelines should help infrastructure administrators seek out these minimum requirements and modify them to accommodate the user concurrency, feature implementation, and operation expectations in your organization.

In this exercise you will explore the MicroStrategy ReadMe for the latest version of the MicroStrategy platform and think about the system requirements that you will outline in your organization's hardware planning standards.

Identify infrastructure requirements

1. Access the ReadMe for the latest version of the MicroStrategy platform: **https://www2.microstrategy.com/producthelp/current/Readme/en-us/content/home.htm**

   Alternatively, you can perform a web search for the ReadMe file for a specific version of the MicroStrategy platform.

Set requirements for the Intelligence Server machine

2. From the ReadMe, in the left pane, expand **Readme>System Requirements>ProductionDeployment>Servers>Intelligence Server**.

3. What is the minimum amount of RAM memory required for the Intelligence Server machine?

4. Based on the features, user concurrency, and performance expectations in your environment, how would you modify the minimum amount of RAM for the Intelligence Server machine in your organization?

5. How would you determine the required hard disk space on the Intelligence Server machine?

6. If you plan to install the Linux operating system on the Intelligence Server machine, which certified product and version would you choose to install?

Set requirements for client machines

You expect that the majority of users in your organization will perform administrative and data analysis tasks through MicroStrategy Workstation. To ensure that users are able to perform their duties efficiently, you must set hardware standards for the machines on which Workstation is installed.

7 From the ReadMe, in the left pane, expand **Readme>System Requirements>ProductionDeployment>Clients>MicroStrategy Workstation**.

8 What is the minimum amount of RAM and disk space required to install MicroStrategy Workstation on a Mac?

9 Based on the operating system, required third-party applications, user workflows, and MicroStrategy platform feature utilization in your organization, how would you modify the minimum hardware requirements in your standards documentation?

Identify certified metadata repository platforms

10 With each platform release, MicroStrategy tests and certifies products with which it integrates. To ensure that your installation performs optimally with third party software, identify the certification level before configuring your software.

11 From the ReadMe, in the left pane, expand **Readme>Platform Certifications**.

12 Which products and versions are Platinum-certified for use as the metadata repository?

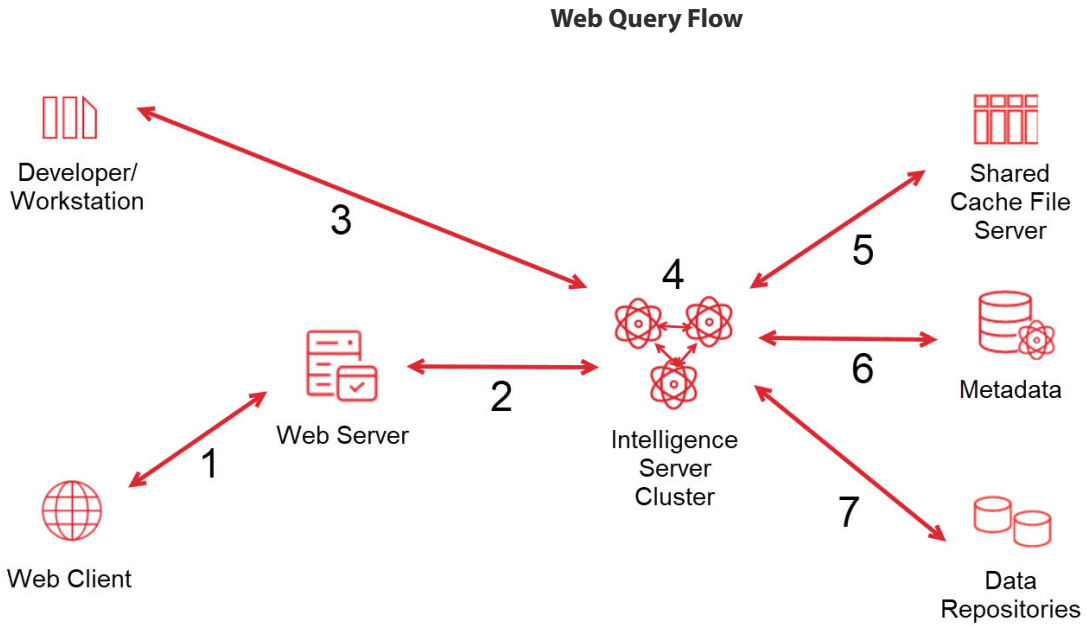
13 Which products are Platinum-certified to be used as the web browsers to use to access MicroStrategy Web?

Establishing a network architecture for your data flow

Your organization's network architecture specifies your network's physical components and their functional organization and configuration. To ensure that the network is secure and capable of transferring data in response to user demands, investigate the required flow of traffic in your network and then develop network architecture standards for administrators to implement as they install and configure the network.

Understanding your network's traffic requirements

To achieve optimal performance with the MicroStrategy platform, its server components must be installed on distinct machines. The network plays an important role in connecting these components. In the following example, the components of a four-tier MicroStrategy system are linked by lines representing the network.



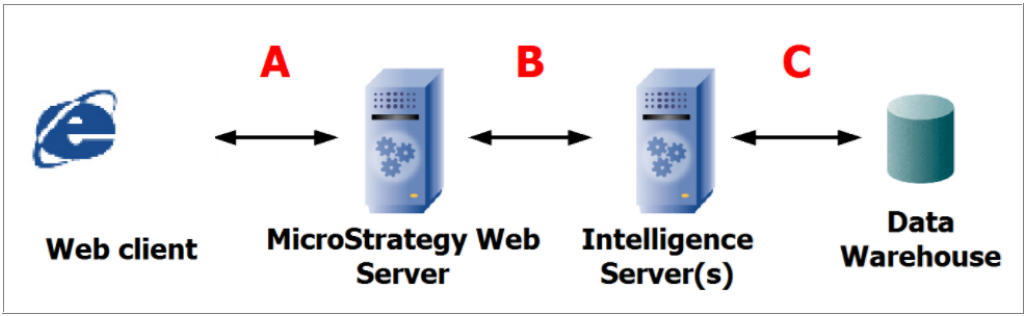
The following table describes the connections in the four-tier system.

Step	Connection	Details
1	HTTP	HTML sent from Web server to client. Data size is small compared to other points because results have been incrementally fetched from Intelligence Server and HTML results do not contain any unnecessary information.
2	TCP/IP	XML requests are sent to Intelligence Server. XML report results are incrementally fetched from Intelligence Server.
3	TCP/IP	Requests are sent to Intelligence Server. (No incremental fetch is used.)
4	TCP/IP	Broadcasts between all nodes of the cluster (if implemented): metadata changes, Inbox, report caches. Files containing cache and Inbox messages are exchanged between Intelligence Server nodes.
5	TCP/IP	Files containing cache and Inbox messages may also be exchanged between Intelligence Server nodes and a shared cache file server if implemented
6	ODBC	Object requests and transactions to metadata. Request results are stored locally in Intelligence Server object cache.
7	ODBC	Complete result set is retrieved from database and stored in Intelligence Server memory and/or caches.

Your network design depends on the type of queries that your users typically run. These queries, in turn, determine the load on the system and the amount of

network traffic between the system components. The network load distribution in your organization likely resembles the following pattern:

Network usage



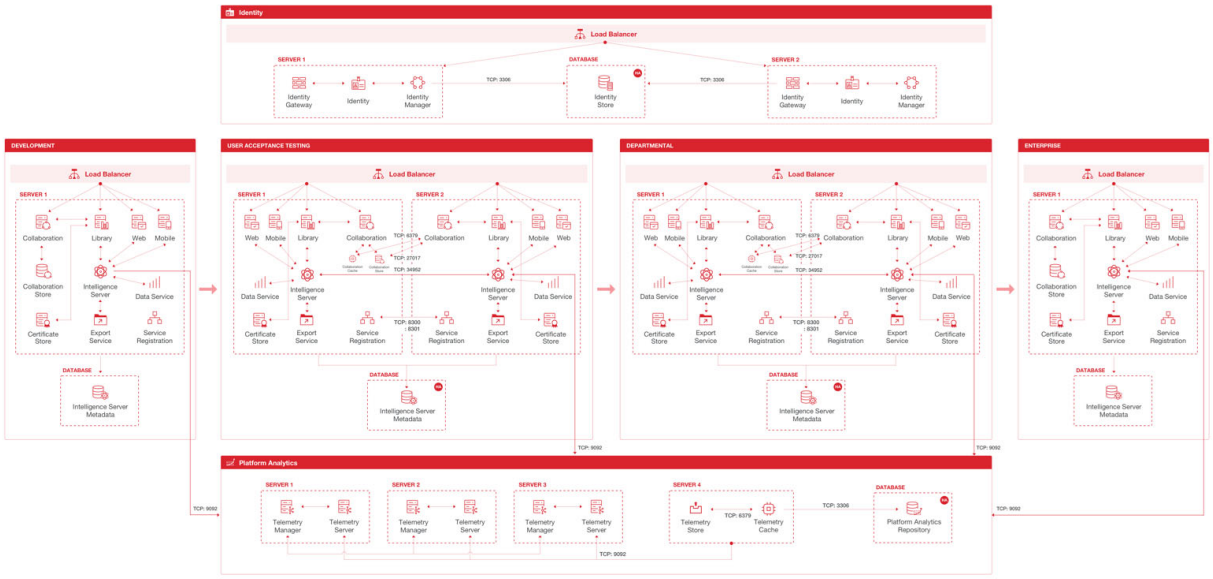
- The largest amount of traffic typically occurs between the data warehouse and the Intelligence Servers (indicated by C in the image above).
- The load between Intelligence Server and Web server (indicated by B) is less than the traffic in C, followed by less traffic between the Web server and the Web browser (indicated by A).
- Incremental fetch size directly impacts the amount of traffic at A.
- Graphics rendering increases network traffic at B.
- The load at C is determined primarily by the number of rows retrieved from the data warehouse. Actions such as sending SQL or retrieving objects from the metadata produce minimal traffic.
- Cached reports do not cause any network traffic at C.

Report manipulations that do not require SQL generation and delivery to the data warehouse (such as pivot, sort, and page-by) produce a similar amount of traffic as cached reports. In contrast, report manipulations that cause SQL to be generated and sent to the data warehouse produce a similar amount of traffic as non-cached reports of the same size.

Creating network architecture guidelines

Once you understand your network's load requirements, you will need to adjust your network bandwidth or change the placement of system components to

accommodate the load. The following example shows the placement of system components within multiple environments.



The following web page contains detailed information about each MicroStrategy component including default installation locations, ports, and protocols:

http://arch.customer.cloud.microstrategy.com/

Keep this information in mind as you develop your network architecture.

To communicate the network's traffic requirement and proposed architecture updates, create guidelines for administrators to follow as they adjust the network architecture.

At a minimum, the standards that you develop for your network architecture should include the following information:

- The physical location of database servers, MicroStrategy Intelligence Server, MicroStrategy Web Server, MicroStrategy Mobile Server and MicroStrategy client machines
- The relative distance between each server and an estimate of the required bandwidth between them. The distance between server components can have an impact on performance. For example, if there is a large distance between the Intelligence Server and the data warehouse, users may experience network delays. To avoid network delays, your standards should employ the following best practices:

Best Practice

  - Place the Web server machines close to the Intelligence Server machines.

    - Place Intelligence Server close to the both the data warehouse and the metadata repository.
    - Use a dedicated machine for the metadata repository.
    - If you have a clustered environment with a shared cache file server, place the shared cache file server close to the Intelligence Server machines.
- A shared and secure network location that contains the MicroStrategy installation files
- Credentials for a service network account that the Platform Administrator uses to:
    - Install and register MicroStrategy Intelligence Server
    - Install and register MicroStrategy Web and MicroStrategy Mobile Servers
    - Install MicroStrategy client products and tools
    - The service network account must contain permissions to:
        - Install or update system level files during the installation
        - Register and run Intelligence Server as a service
        - Access the Home, Log, Inbox, Cube, and Cache storage directories
- A service (network) account for the Database Architect to install and configure database servers
- A network account used to run Intelligence Server. This account requires privileges to read from and write to the Home, Log, Inbox, Cube, and Cache storage directories.

Safeguarding data transmissions in your network

The network security architecture encompasses the collective measures that protect the usability and integrity of the network from illegitimate use, malicious threats, and attacks. To ensure data security on your organization's network, create standards to address the following security aspects of your network architecture:

- Establish secure communications between server and client components through tunneling, VPN, SSL/TLS, certificates, and so on.
- Identify and enable the ports required for MicroStrategy services and communications processes.

- Document the ports used for MicroStrategy services and third-party software in your environment. The following table shows the default port numbers used by a few MicroStrategy services and processes.

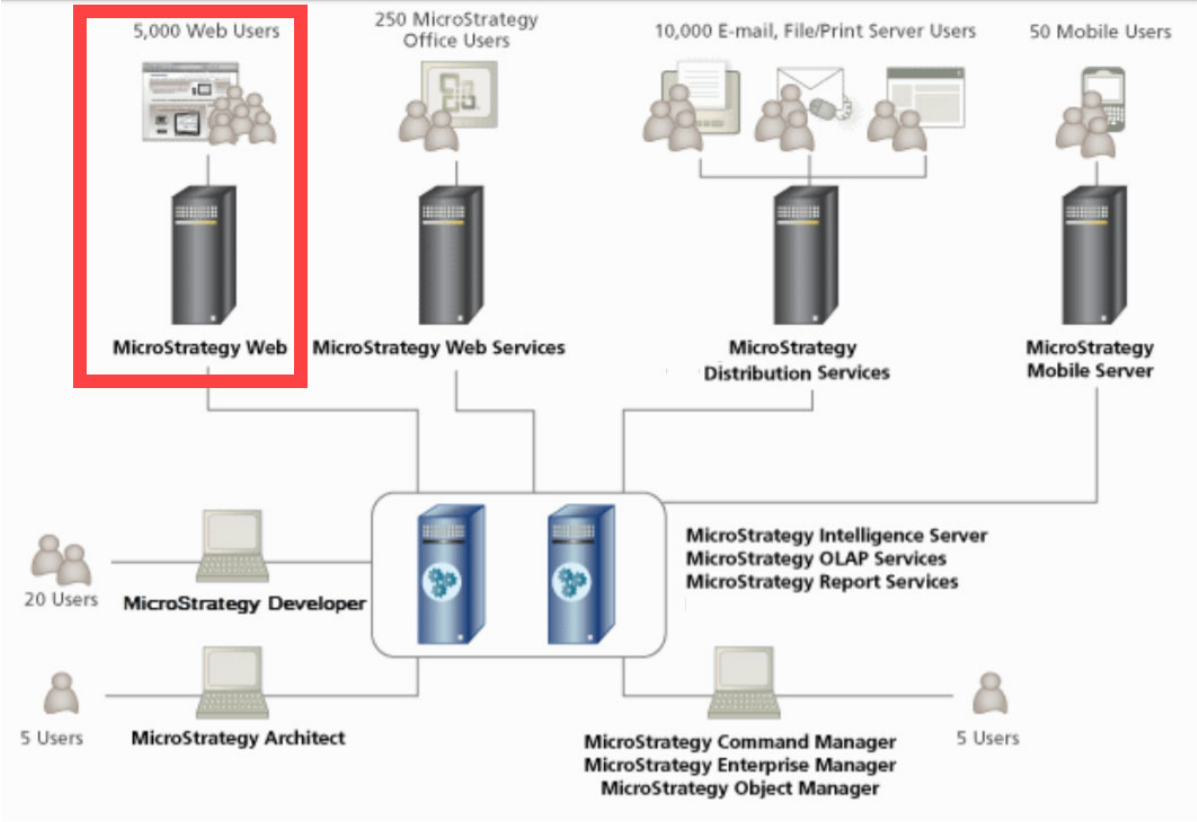
MicroStrategy service/process	Default port
Apache Kafka	9092
Apache Tomcat (Dossier Web)	8080
Apache Tomcat (Identity Management System (IDM))	443/1443
Apache Zookeeper	2181
MongoDB	27017
Redis	6379
MicroStrategy Collaboration/Realtime Service	3000
MicroStrategy PDFExport Service	20100
MicroStrategy REST Server	7070
MicroStrategy Intelligence Server	34952

Which secure communication protocols do you employ between servers and clients in your organization?

Facilitating web-based analytics delivery

In a MicroStrategy system, web server configuration components facilitate the storage, processing, and delivery of content over the web to client machines. The

following image shows the MicroStrategy Web component in the context of a medium-sized production deployment.



To ensure that content is delivered with adequate performance and integrity, create standards to help administrators in your organization install and configure appropriate web server components.

Use the recommended web server specifications outlined in the Installation and Configuration Guide, and define specifications for your organizations based on your unique requirements. Your web server configuration guidelines might include instructions to perform the following:

Best Practice

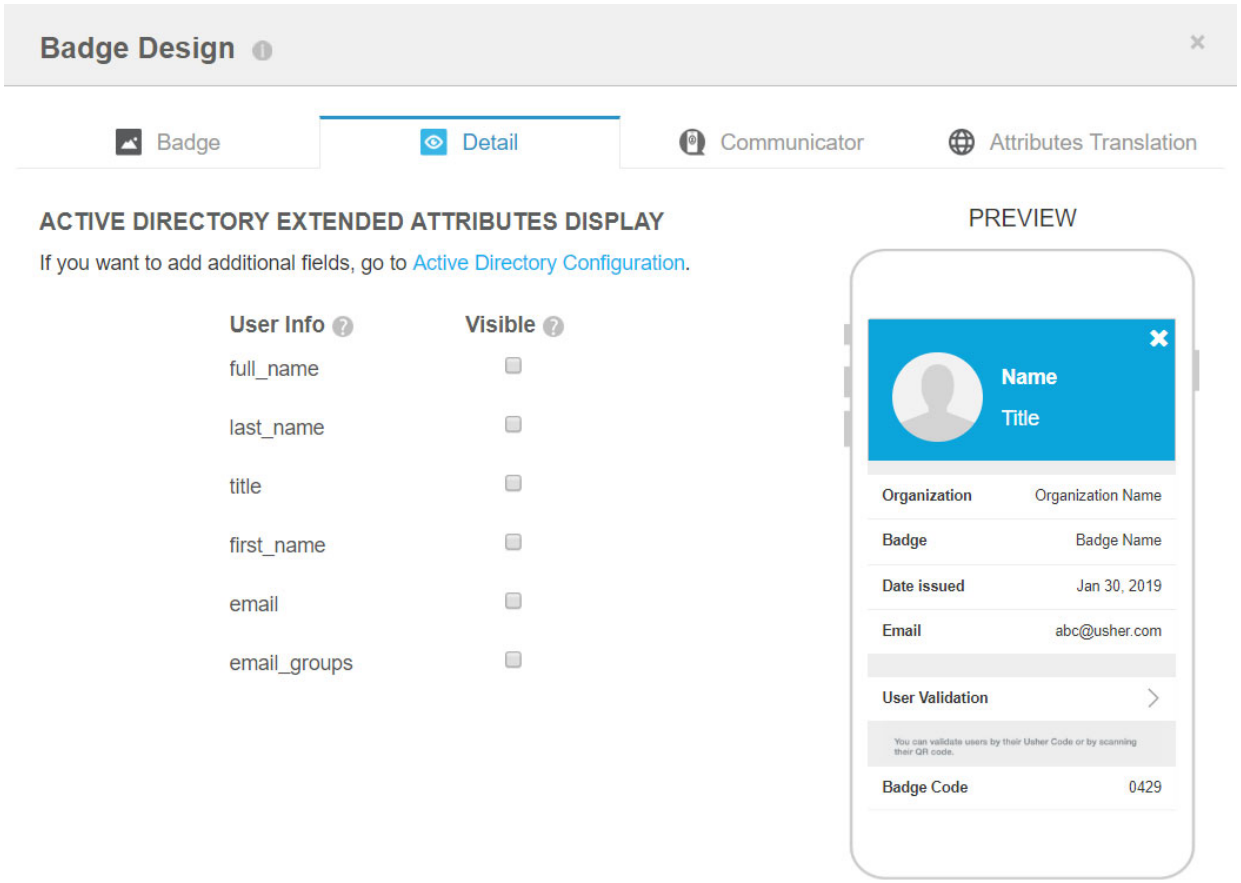
- Install and configure certified versions of the Web Application Server and the Portal Server.
- Based on your organization's preferred maps provider, work with your Platform Administrator to configure integration with third-party mapping services such as ArcGIS, ESRI or Google Maps.
- To achieve optimal performance based on user concurrency and peak utilization, configure clustering and load balancing for MicroStrategy Web Server and MicroStrategy Mobile Server.

- Install and set up a certified version of Java Developer Kit (JDK) and Java Virtual Machine (JVM).
- To leverage the latest features and security configurations, apply the latest patches and updates to the web server configuration components.

Maintaining a central repository for user information

Directory services applications store information about users, groups, files, printers and other resources. These applications can be used as a source for user information to import into the MicroStrategy platform when creating and maintaining users.

You can import user information from Active Directory to create users in the MicroStrategy Analytics platform. This enables you to manage user information in a single location instead of recreating user data in multiple applications.



To streamline user creation in MicroStrategy and maintain a single source of user data, create guidelines to help administrators establish and maintain a directory services application like Microsoft Active Directory. Perform the following tasks as you establish your standards:

- Work with the Platform Administrator to identify the authentication mode used to access the MicroStrategy platform.
- Establish a third-party authentication server such as LDAP, Kerberos, Siteminder, Tivoli, Ping Federate, Oracle Access Manager, or SAML. In your standards, include installation and maintenance instructions for the server.
- Work with the Platform Administrator to connect MicroStrategy to the third-party authentication server. Document instructions to establish connectivity.
- Establish a schedule to update and apply patches to the selected directory services application.

Do you use a directory services application in your organization?

What benefits have you experienced by maintaining users in a central repository?

With what applications does your directory services interact?

Exercise 3.3: Accessing your cloud environment

The MicroStrategy environment you will use in this class is housed in the cloud through MicroStrategy Cloud. The cloud environment allows you to access a fully functional MicroStrategy installation through your web browser.

In this exercise, you will access the Windows machine in your cloud environment and launch MicroStrategy Developer. The login credentials and other information required to access your environment are included in the MicroStrategy Cloud email.

Access the cloud environment

1. On your local machine, in the MicroStrategy Cloud email, click **Access MicroStrategy Platform**.
2. In the **User name** and **Password** boxes, enter the login credentials provided in the MicroStrategy Cloud email. Click **Login**.

   The MicroStrategy Cloud landing page is displayed.
3. On the landing page, hover over **Remote Desktop Gateway** and click **Launch**. The Remote Desktop Connection window is displayed.
4. In the Clipboard message at the top of your browser, click **Allow** to enable copying-and-pasting between your local machine and the remote machine.
5. In the Apache Guacamole window, in the **Username** and **Password** boxes, type the user name and password listed in the MicroStrategy Cloud email.
6. Under All Connections, click **Developer Instance RDP**. You are now connected to the Windows machine in your cloud environment.

Exercise 3.4: Installing and configuring Active Directory

Because InfiniRec is a startup and the employee base is currently minimal, they have tracked user information in an Excel spreadsheet. The company has received a new round of funding and is in the process of hiring hundreds of employees. Management wants to get away from the spreadsheet and create a central repository for user information.

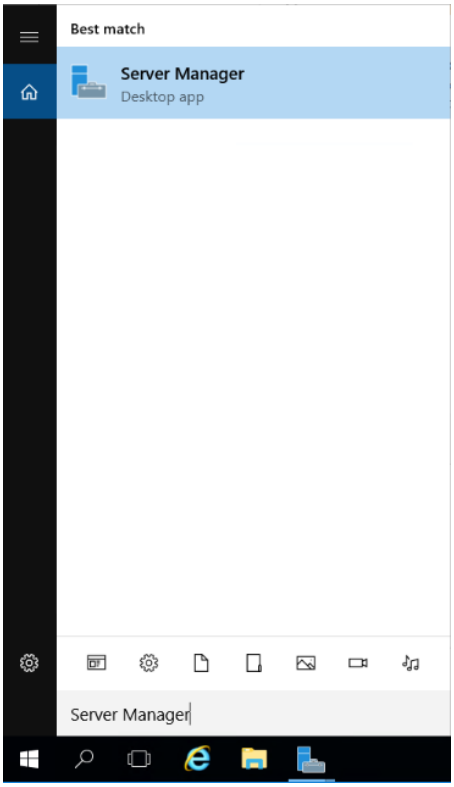
In this exercise you will install and configure Active Directory on a Windows server. As you perform the exercise, think about a central location in your organization where you would install Active Directory, and identify the configuration steps that you would include in your standards documentation.

Install Active Directory

1 Access the RDP connection to the Windows machine in the cloud.

2 From the task bar, Click **Search Windows.**

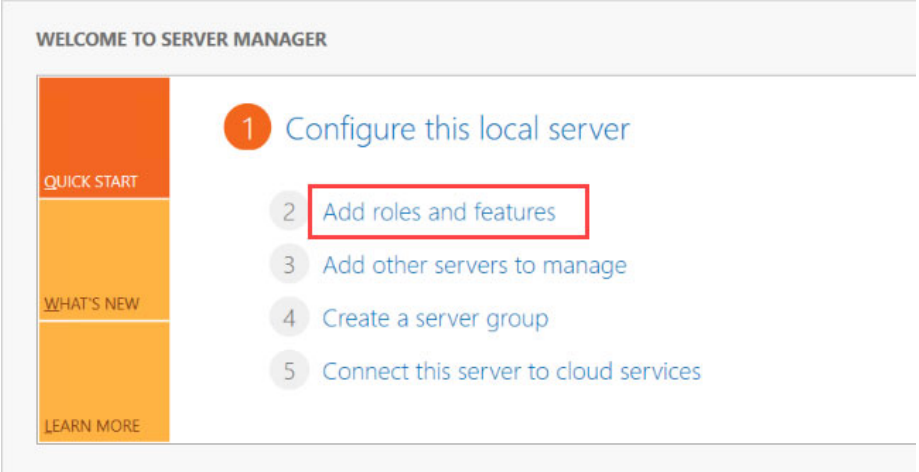
3 In the **Search** box, type **Server Manager**.



4 From the search results, click **Server Manager**.

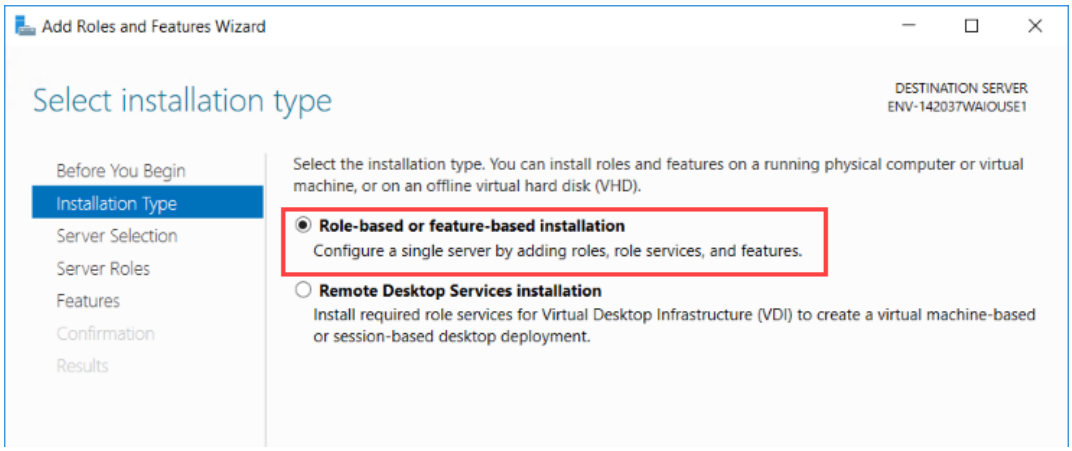
Install Active Directory

5 In the Welcome to Server Manager window, click **Add roles and features**.

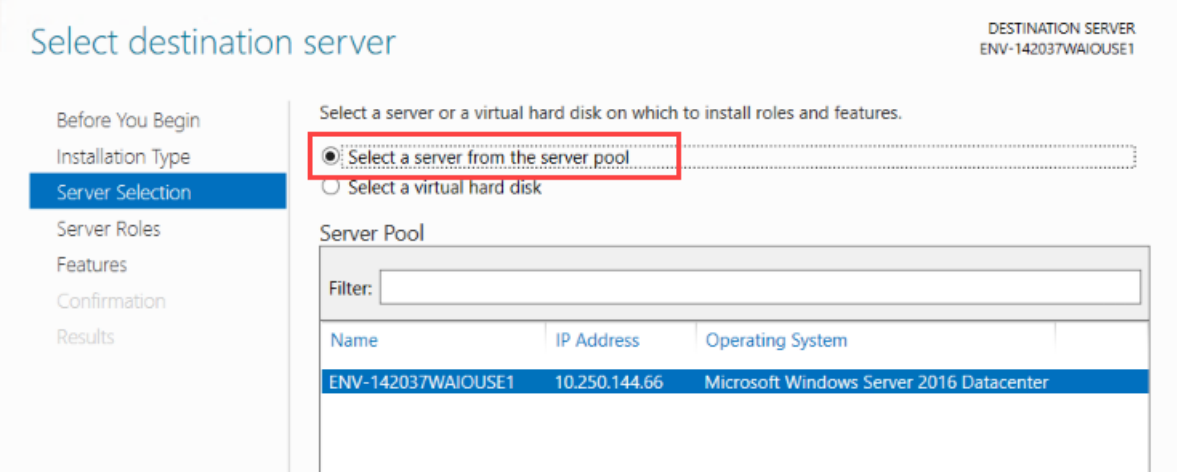


6 In the Before You Begin window, click **Next**.

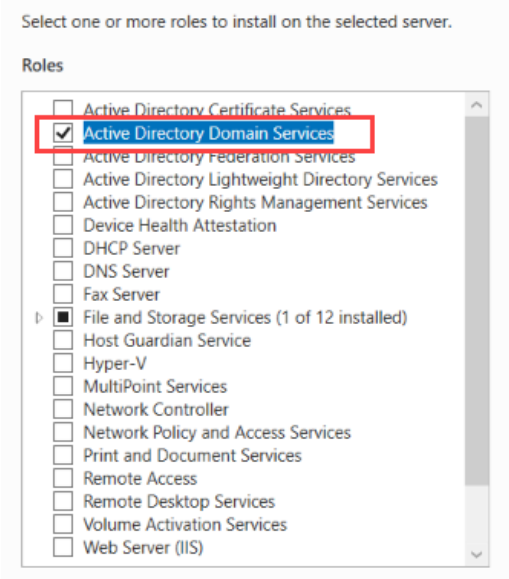
7 In the Installation Type window, ensure that **Role-based or feature-based installation** is selected and click **Next**.



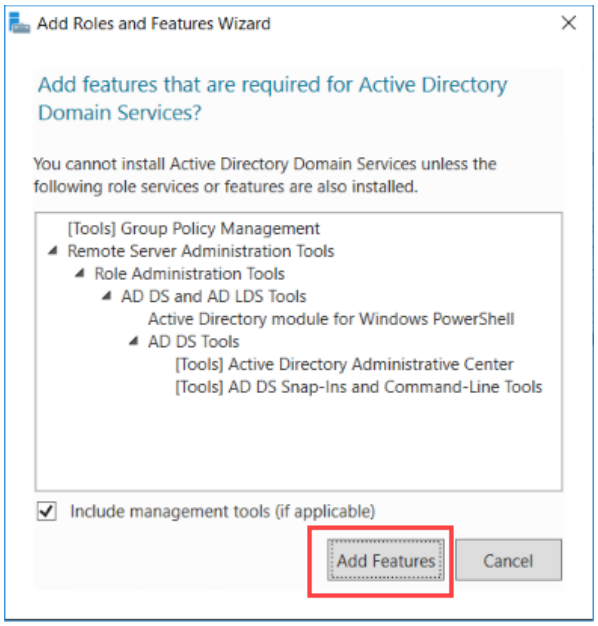
8 In the Server Selection window, ensure **Select a server from the server pool** is selected and click **Next**.



9 In the Server Roles window, in the Roles area, click **Active Directory Domain Services**.



10 In the Add features that are required for Active Directory Domain Services window, click **Add Features**.

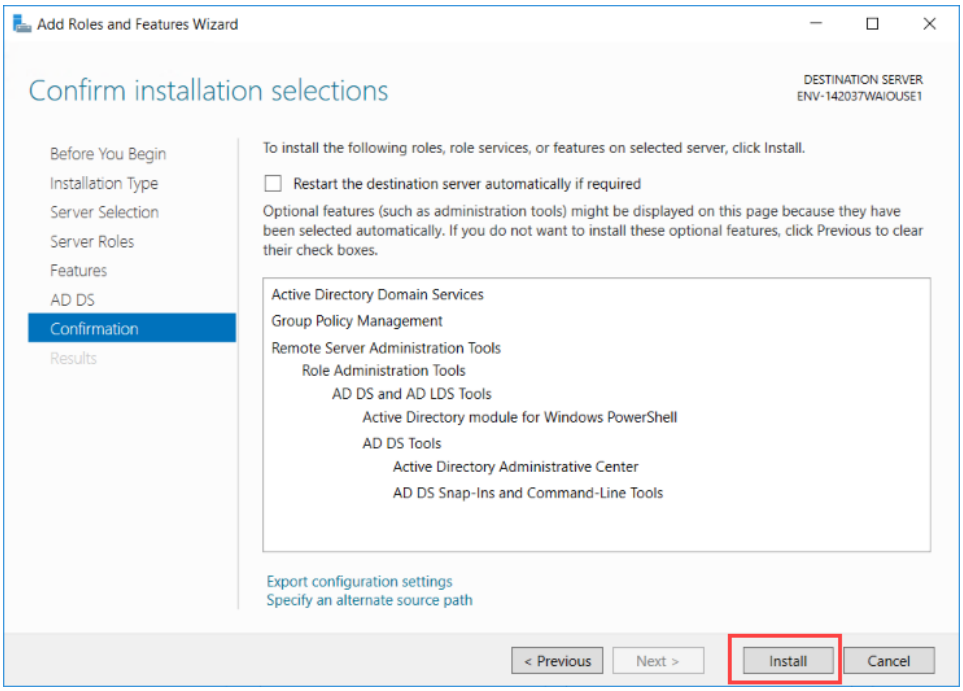


11 Click **Next**.

12 In the Select features window, click **Next**.

13 In the Active Directory Domain Services window, click **Next**.

14 In the Confirm installation selections window, click **Install**.

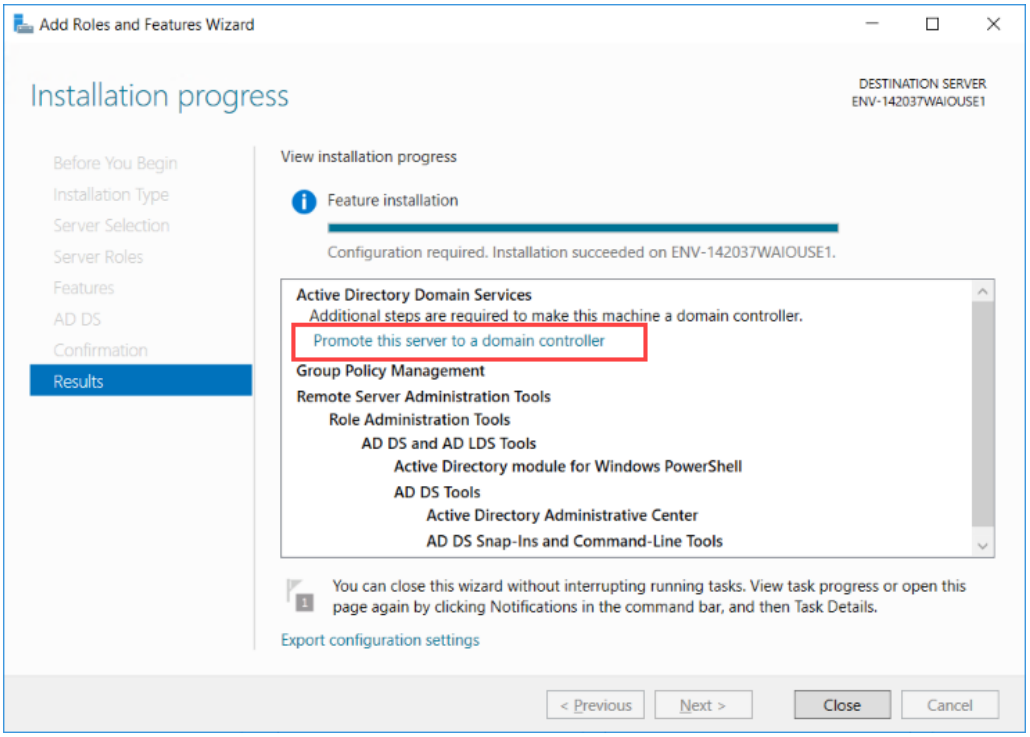


15 Leave the Add Roles and Features Wizard window open.

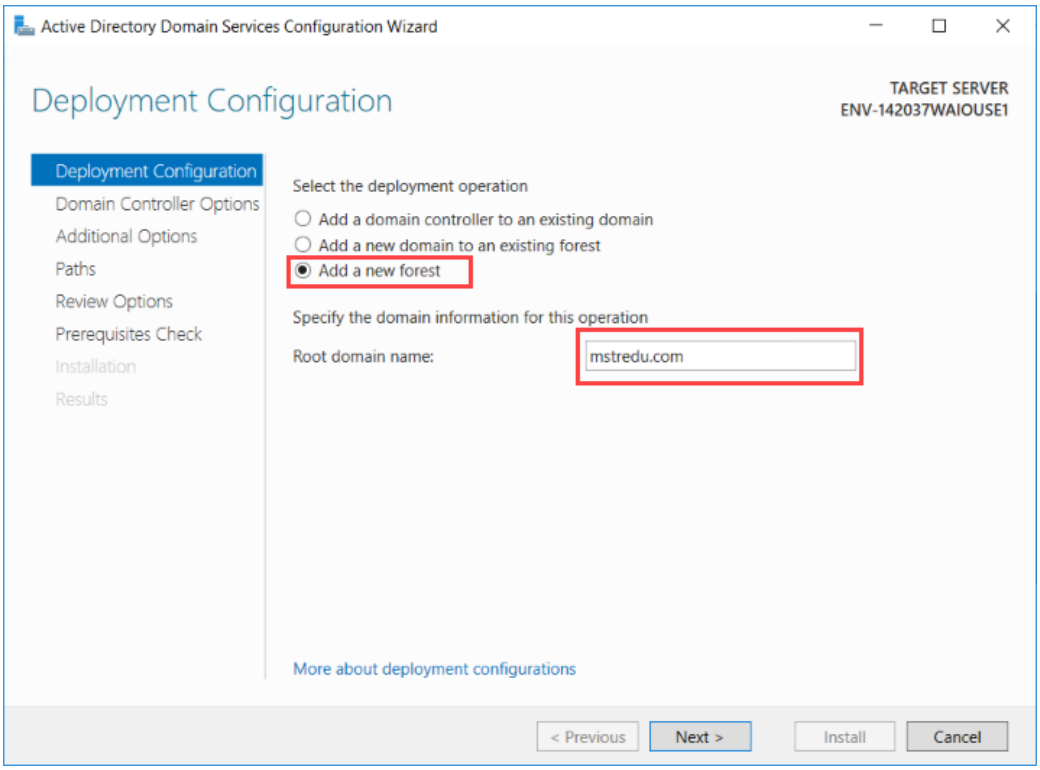
Now that you have installed Active Directory on the server, you are ready to configure the service details.

Configure Active Directory

1 After the installation finishes, click **Promote this server to a domain controller**.

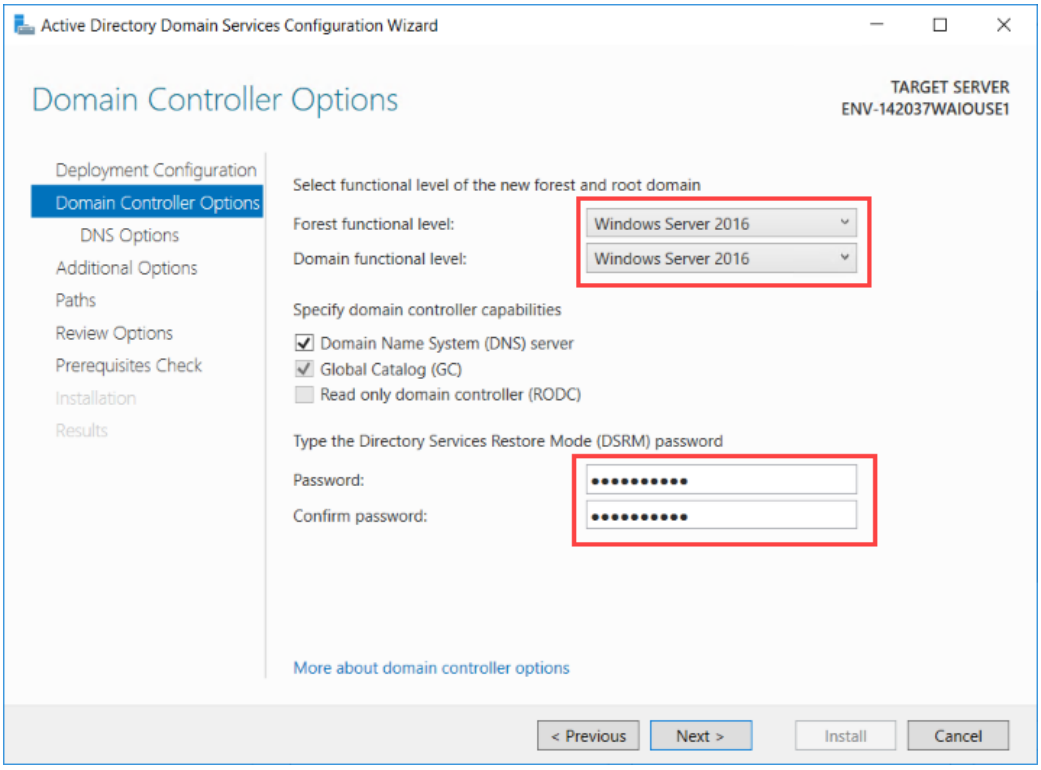


2 In the Deployment Configuration window, select **Add a new forest**.



3 In the **Root domain name** box, type **mstredu.com** and click **Next**.

4 In the **Forest functional level** drop-down list, ensure **Windows Server 2016** is selected.



5 In the **Domain functional level** drop-down list, ensure **Windows Server 2016** is selected.

6 In the **Password** and **Confirm password** boxes, type **EduCloud1!** and click **Next**.

7 In the DNS Options window, click **Next**.

8 In the Additional Options window, wait for the **NetBios domain name** box to be populated with **MSTREDU** and then click **Next**.

9 In the Paths window, click **Next**.

10 In the Review Options window, click **Next**.

11 In the Prerequisites Check window, wait for the check to finish and then click **Install**.

Log back in to the server

12 The server restarts and the RDP connection is disconnected. Wait about a minute and then click **Reconnect** in the RDP browser window.

13 An incorrect user message is displayed. Click **OK**.

14 As part of the Active Directory configuration, you modified the domain name for this server to mstredu. You must now sign in to the new domain. To do this, in the **User Name** box, type **mstredu\mstr**. In the **Password** box, type the password from your MicroStrategy Cloud email.

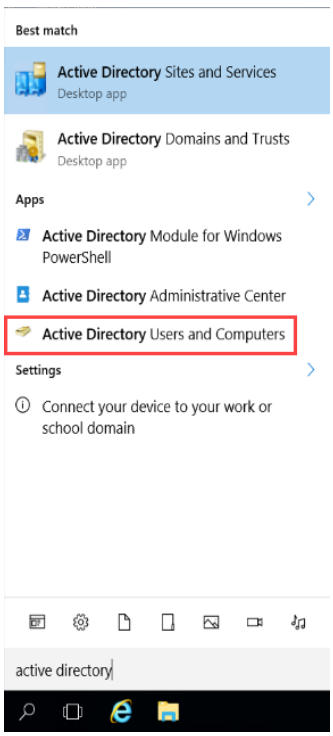
Exercise 3.5: Importing users into Active Directory

You can add users to Active Directory and modify their information through a variety of methods. For example, you can write a script that imports users from a file or manually enter individual users. The user import methods you employ in your organization depend on how you currently store users, the number of users that need to be imported, and so on.

In this exercise, you will import users into Active Directory using a script that pulls user information from a csv file. As you create users, think about how you plan to create users in your own environment and identify guidelines that you plan to establish in your organization.

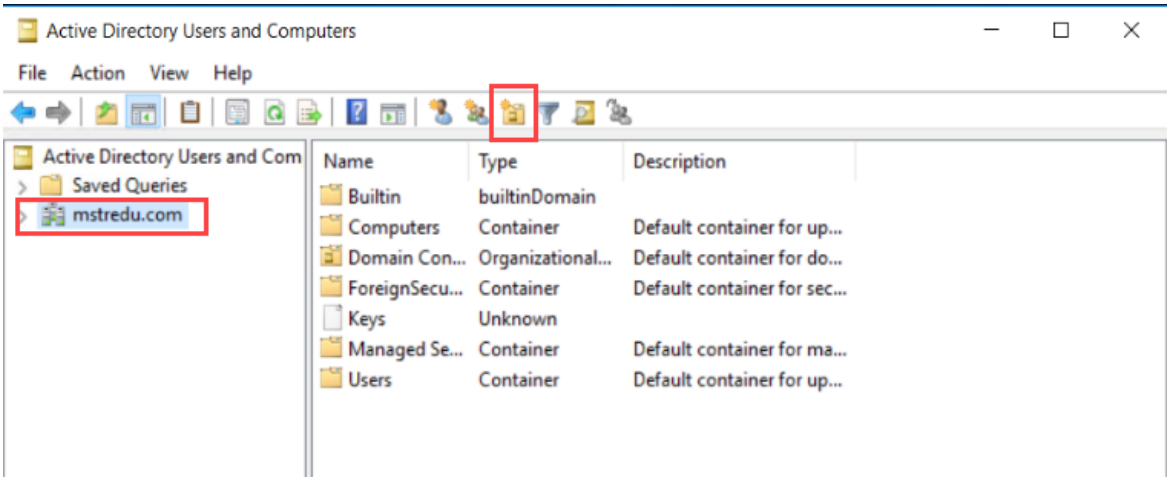
Import users into Active Directory

1. Access the RDP connection to the Windows machine in the cloud.
2. From the task bar, in the **Search** box, type **Active Directory Users and Computers**.
3. From the search results, click **Active Directory Users and Computers**. The Active Directory Users and Computers window opens and mstredu.com is displayed in the left panel.



4 In the left panel, click **mstredu.com**.

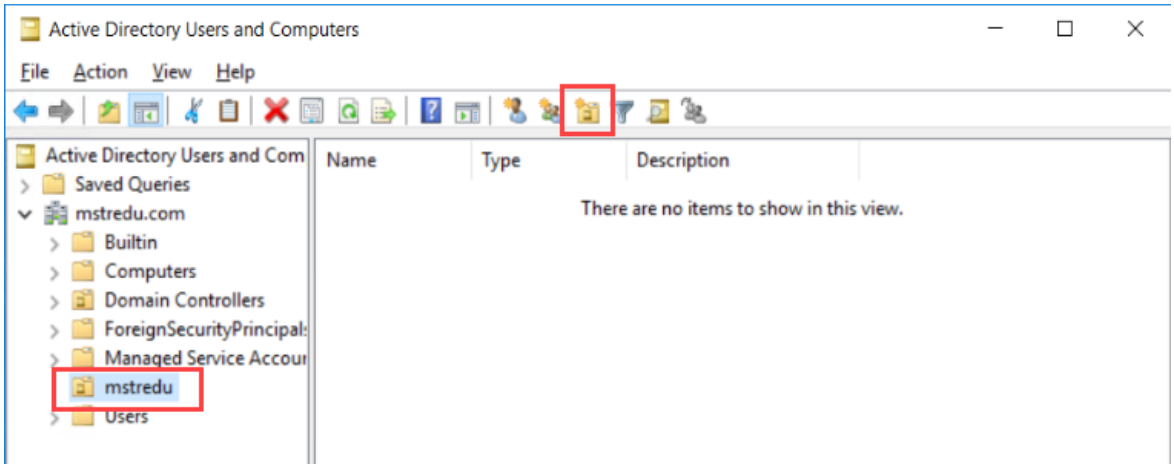
5 Click **Create a new organizational unit in the current container**.



6 In the **Name** box, type **mstredu** and click **OK**.

7 In the left pane, click the **mstredu** folder you just created.

8 At the top of the screen, click **Create a new organization unit in the current container**.



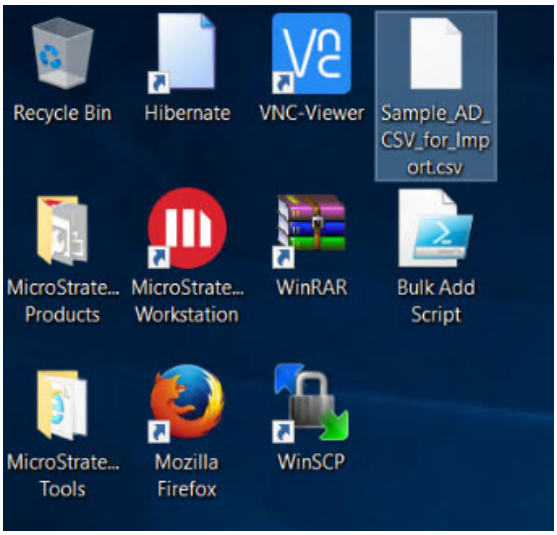
9 In the **Name** box, type **Users** and click **OK**.

Import users through a script

10 Copy the following scripts, provided by your instructor, to the desktop of the Windows machine in the cloud.

- **Bulk Add Script.ps1**

- **Sample_AD_CSV_for_Import.csv**



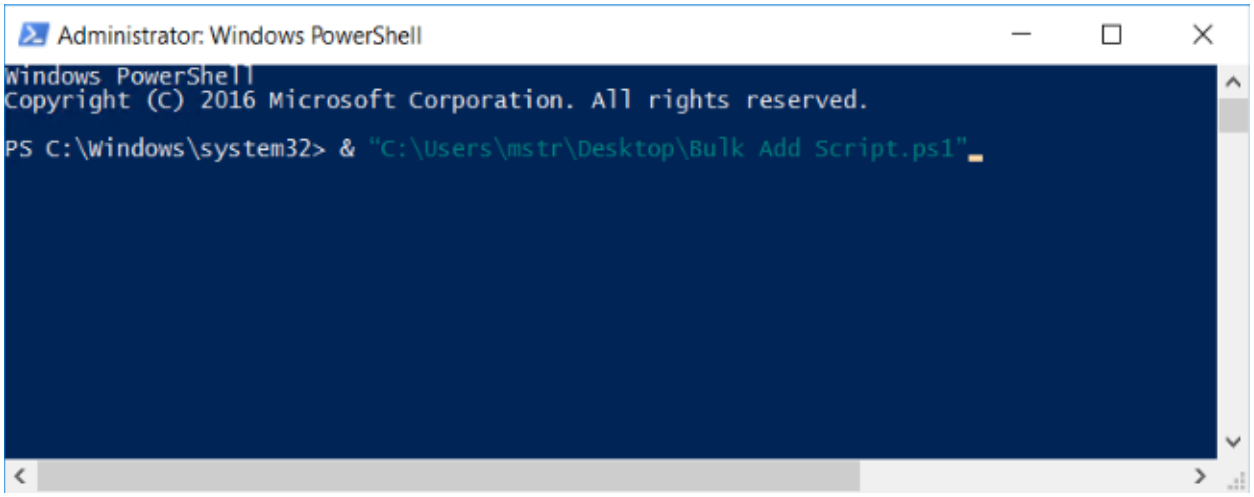
To do in the web-based Apache Guacamole client, drag and drop the files from your local machine to the remote desktop. From File Explorer, open **This PC\Guacamole Filesystem on Guacamole RDP** and drag the files to the desktop.

11 From the task bar, in the **Search** box, type **PowerShell**.

12 From the search results, right-click **Windows PowerShell** and select **Run as Administrator**.

13 In the User Account Control window, click **Yes**.

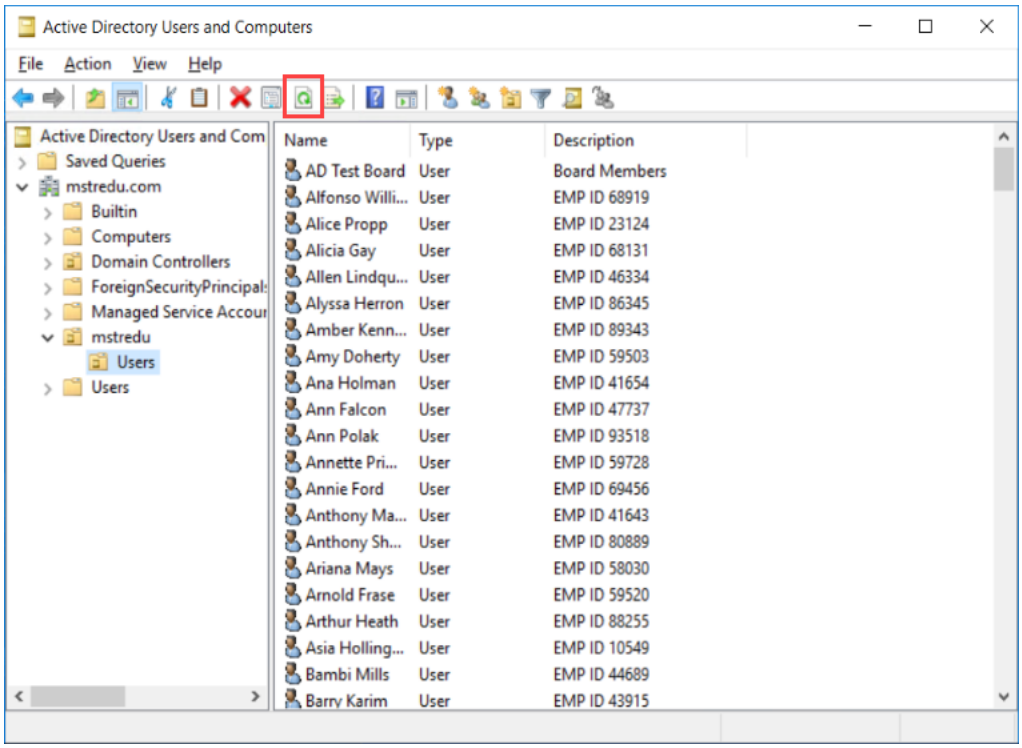
14 In the PowerShell window, type **& "C:\Users\mstr\Desktop\Bulk Add Script.ps1"** and hit **Enter**.



After a few moments, the script transfers the users in the Sample_AD_CSV_for_import file into the mstredu\Users container in Active Directory.

15 From Active Directory Users and Computers, in the left panel, expand **mstredu.com**, expand **mstredu**, and click **Users**.

16 At the top of the screen, click **Refresh**.



The users imported from the csv file are displayed in the mstredu\Users container.

Now that you have imported users from a file, you will learn to manually create a new user and enter identifying information.

Exercise 3.6: Manually adding users to Active Directory

Now that InfiniRec's user base is in a central repository, you want to identify a workflow to enter users into the system as they are introduced to the organization.

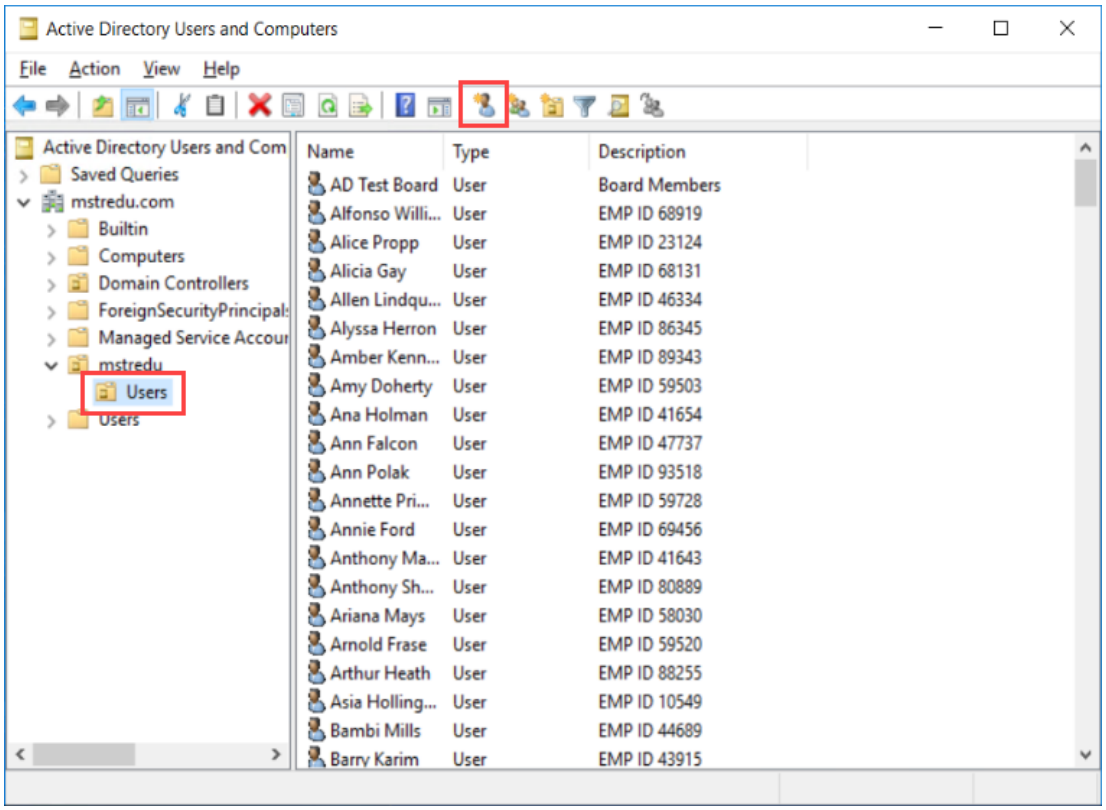
Once Active Directory has been installed and configured, and your users have been imported, you will likely need to add new users as they enter your organization. You will also need to modify user information as users update phone numbers, change tiles, move to different departments, and so on.

In this exercise, manually add a user and update their information. As you manually modify your user, think about the Active Directory workflow you will deploy in your organization, and the restrictions you plan to place on user modifications.

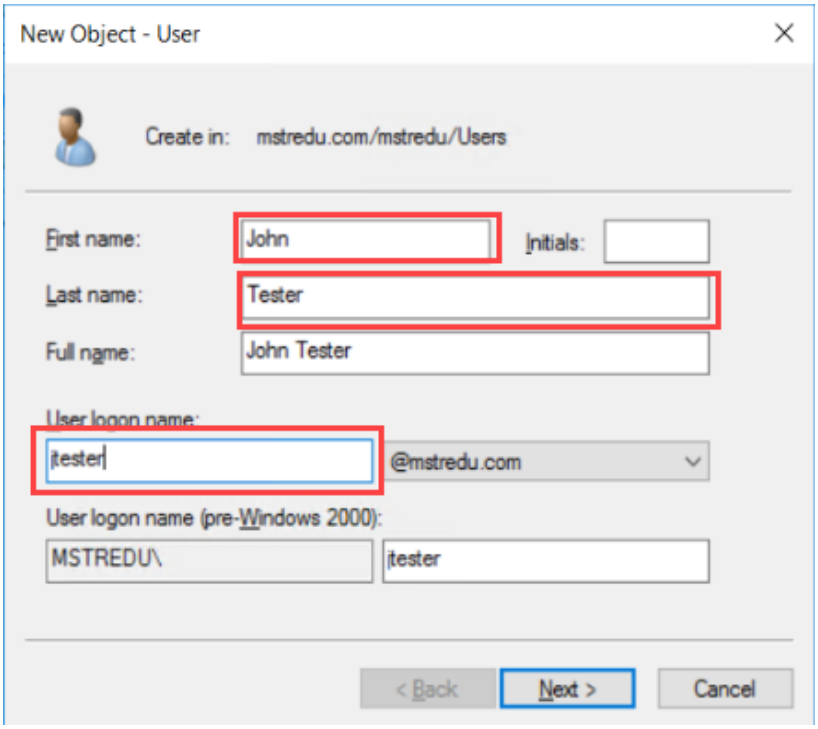
Manually add a new user to Active Directory

1. Access the RDP connection to the Windows machine in the cloud.
2. From the task bar, in the **Search** box, type **Active Directory Users and Computers**.
3. From the search results, click **Active Directory Users and Computers**. The Active Directory Users and Computers window opens and mstredu.com is displayed in the left panel.
4. In the left panel, expand **mstredu.com**, expand **mstredu**, then click **Users**.

5 At the top of the screen, click **Create a new user unit in the current container**.



6 In the New Object - User window, type the following information:



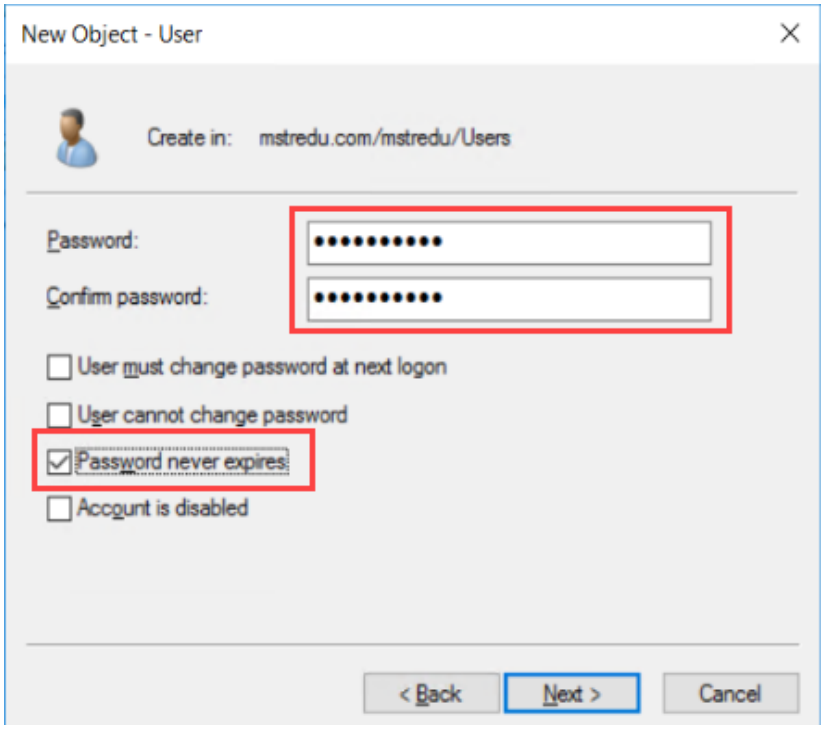
a In the **First Name** box, type ***your first name***.

b In the **Last Name** box, type ***your last name***.

c In the **User logon name** box, type ***a user ID***.

7 Click **Next**.

8 In the **Password** and **Confirm Password** fields, type **EduCloud1!**

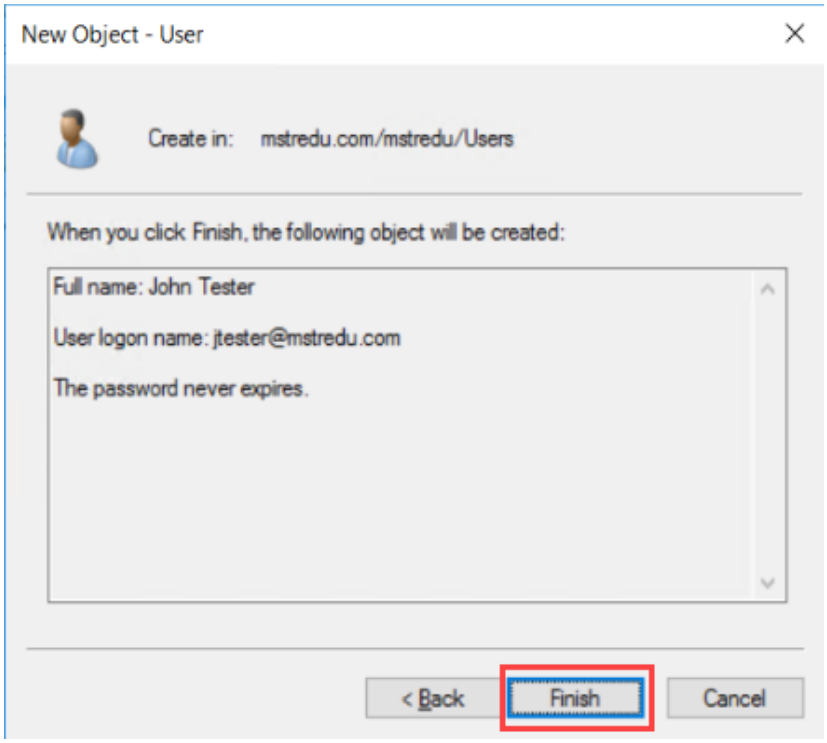


9 Click the **Password never expires** check box.

10 In the Password window, click **OK**.

11 Click **Next**.

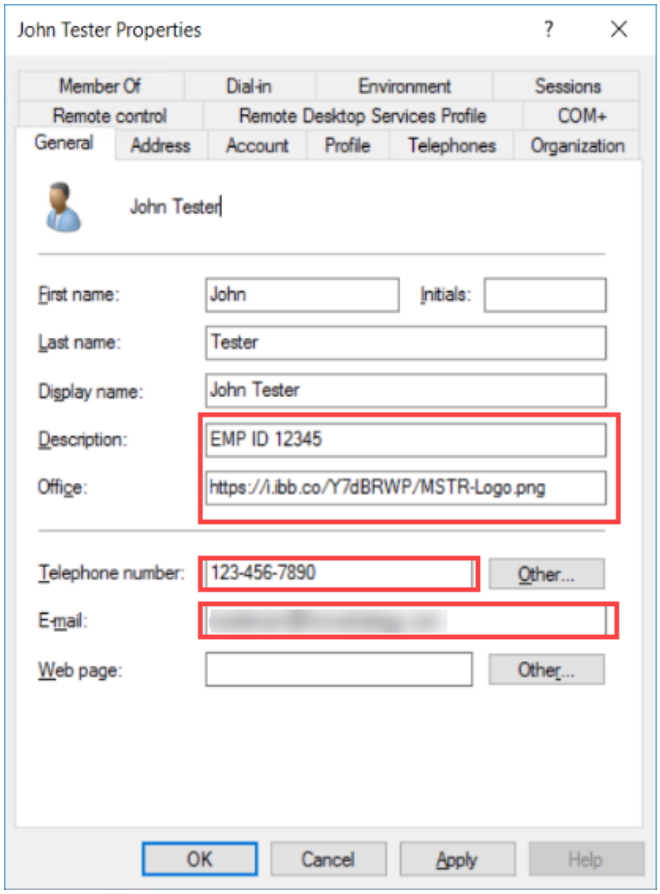
12 Review the summary and click **Finish**.



Update user information

13 In Active Directory Users and Computers, double-click the user you just created.

14 In the **General** tab, type the following information:



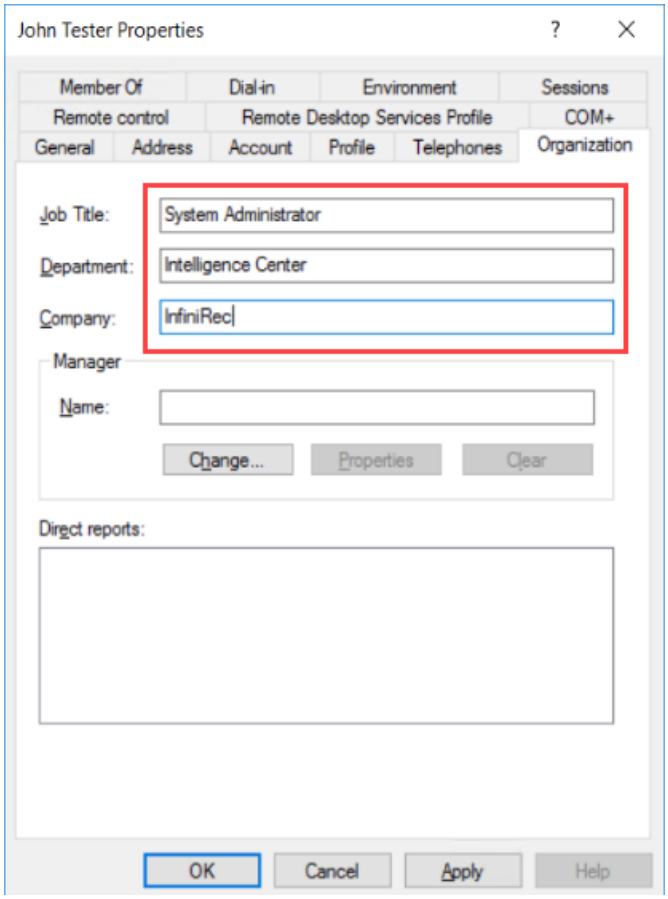
a In the **Description** box, type **EMP ID 12345**.

b In the **Office** box, type **https://i.ibb.co/Y7dBRWP/MSTR-Logo.png**

c In the **Telephone number** box, type **123-456-7890**

d In the **E-mail** box, type ***your email address***.

15 Click the **Organization** tab, and type the following information:



a In the **Job Title** box, type ***your title***.

b In the **Department** box, type ***your department or division***.

c In the **Company** box, type ***your organization's name***.

16 Click **OK**. You created a new user and updated several attributes.

Establishing a repository for MicroStrategy system files

The server file system enables storage, organization, and retrieval of critical files required to install, configure, and optimize the MicroStrategy platform. The following image shows an example layout of a Linux file system.

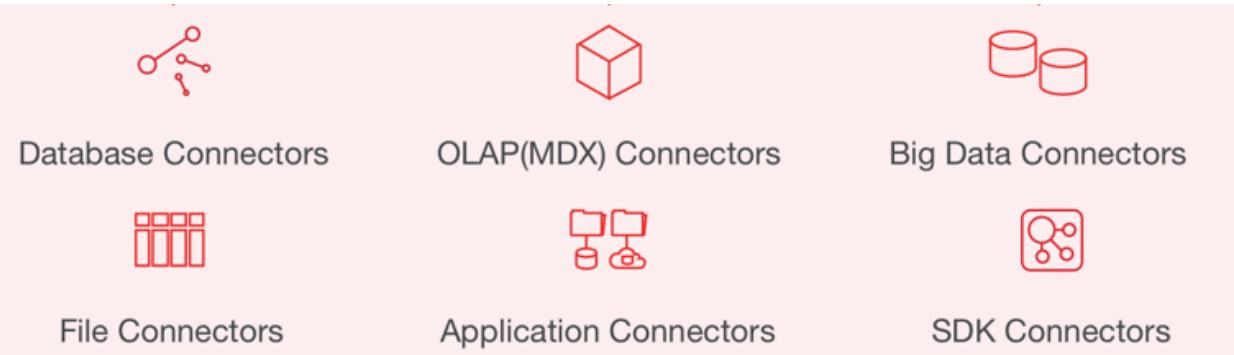
To ensure that MicroStrategy files are readily available to the appropriate parties or accounts, create guidelines to help administrators perform the following file management practices.

- A storage map like the preceding image can help you make decisions about an ideal storage location for shared files.
- Establish a shared location to store MicroStrategy product installation files. Access to the shared location should be provided to the Platform Administrator and any other parties who are responsible for installing and configuring the MicroStrategy platform.

- Create a location for cache, history and cube files. To do this, mount a disk to make it accessible through the server's operating system, and then create a service account to access the cache and cube files.

Connecting to data sources

ODBC drivers establish connectivity between the MicroStrategy platform and your data sources.



Create standards in your organization to help administrators identify the appropriate data source connectors and drivers to use. For example, you might include the following guidelines for installing appropriate connectors.

- Installing and updating certified ODBC drivers not shipped with MicroStrategy products, including installing any dependent files and libraries
- Installing and updating native data source connectors, including installing any dependent files and libraries

The connectivity practices you establish in your organization depend on several factors including the type of data sources you employ, your chosen operating system, and so on. The connectivity standards you establish in your organization might include guidelines to perform the following functions:

Best Practice

- Establish an ODBC connection by leveraging MicroStrategy-shipped drivers through Connectivity Wizard.
- If the drivers you want to use are not shipped with MicroStrategy, establish an ODBC connection by leveraging drivers not shipped with the MicroStrategy platform. Based on your operating system, this process requires one of the following workflows:
  - To establish connectivity in a Windows environment, use MicroStrategy Connectivity Wizard or Microsoft ODBC Administrator.

    - To establish connectivity on a Linux environment, modify the odbc.ini, odbc.ini.example, and ODBC.sh files.
    - Specify the exact parameters that must be configured for each database type in the ODBC.ini file.
- Update ODBC connectivity information to accommodate configuration changes. For example, changes in the database server name, IP address, or port numbers should be reflected in the ODBC connectivity parameters.
- Configure native connectivity to Big Data and related data sources.

Exercise 3.7: Establishing an ODBC connection through the ODBC.ini file

Now that the infrastructure has been procured and established, InfiniRec is ready to connect to data sources and begin extracting insights. InfiniRec has several data sources that have compatible drivers provided by MicroStrategy. However, there are a couple of data sources that do not have supported drivers in MicroStrategy. In this exercise, modify MicroStrategy's ODBC.ini file to accommodate all data source connections.

ODBC connections enable communication between the MicroStrategy platform and your data sources and metadata warehouse. For optimal performance, use Connectivity Wizard to connect to data warehouses using out-of-the-box drivers provided in the MicroStrategy installation.

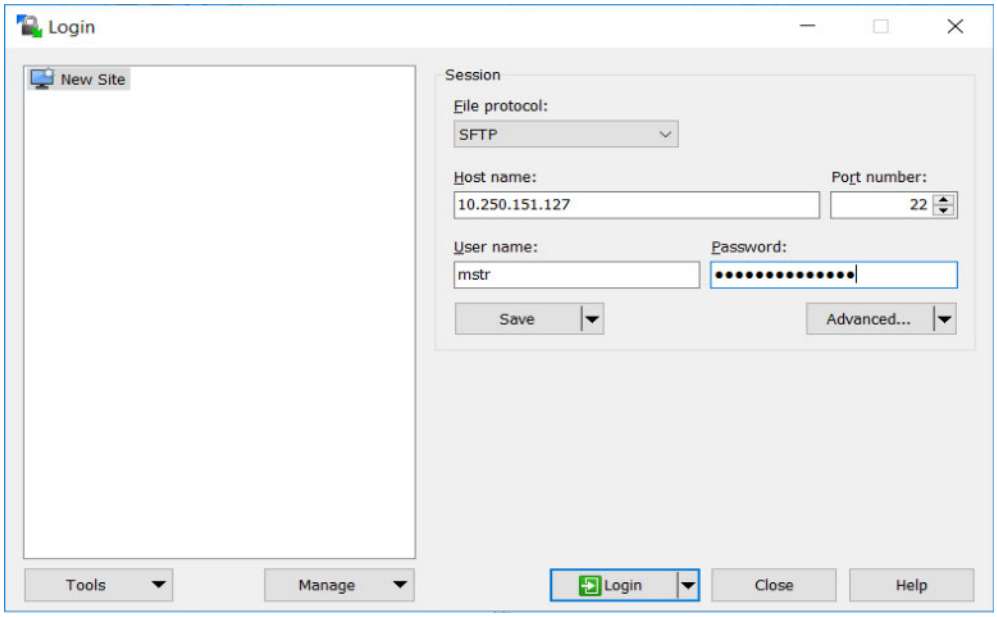
Alternatively, if the desired driver is not included with MicroStrategy, you can create a connection by entering the required information in the odbc.ini file. Each database has a uniques set of parameters that can be configured in the odbc.ini file. In this exercise, configure a connection to a PostgreSQL database.

Create an FTP session between the Linux and Windows servers

1 On the remote Windows desktop, double-click **WinSCP**.

2 In the Login - WinSCP window, in the **Host Name** box, type the Intelligence Server IP address listed in the **hosts.txt** file.

3 In the User name and Password boxes, type the User name and password provided in the MicroStrategy Cloud email.



4 Click **Save**, click the **Save password** check box and then click **OK** to save your session information for future use.

5 Click **Login**.

If a warning window is displayed, click **Yes**.

6 In the Authentication Banner window, click **Continue**.

7 In the WinSCP window, in the drop down list at the top of the right pane, select **/<root>** and then browse to **\opt\mstr\MicroStrategy**.

View example configuration parameters for various databases

8 In the right pane, double-click **odbc.ini.example**. The example file opens and displays possible configuration parameters for each database type.

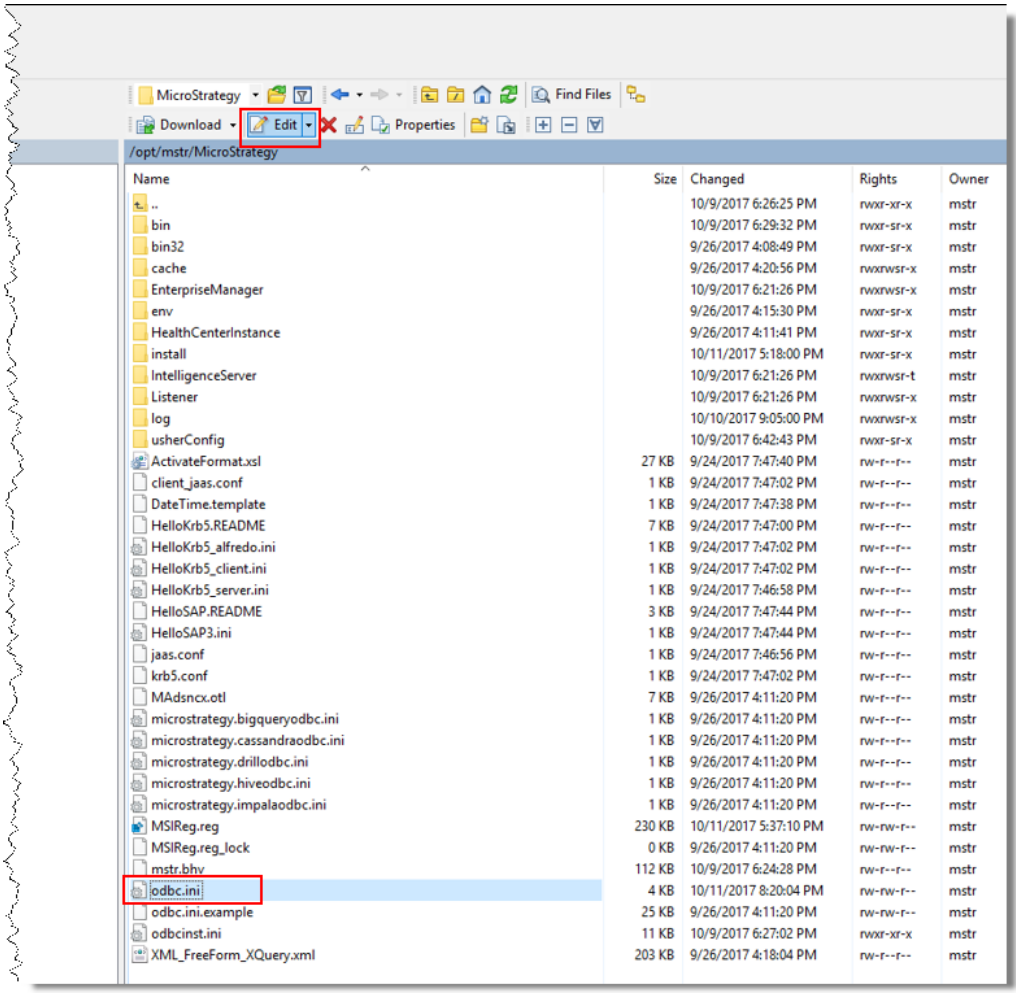
9 Hit **Ctrl** + **f** and search for **PostgreSQL**.

10 Click **Next** until you see the connection parameters under [PostgreSQL]. You can use some or all of these parameters to create a connection to your PostgreSQL database.

11 Close the **odbc.ini.example** file without saving.

Add a new database connection

12 In the WinSCP window, in the right pane, select **odbc.ini** and on the top of the right pane, click **Edit**.



The odbc.ini file displays in the text editor.

13 Towards the top of the odbc.ini file, under [ODBC Data Sources] but above [ODBC], type the following to create the MY_WH data source and use the MicroStrategy PostgreSQL ODBC driver to establish the connection:

MY_WH=MicroStrategy ODBC Driver for PostgreSQL Wire Protocol

14 At the end of the odbc.ini file, type the following information to specify the details for the MYWH data source in a PostgreSQL database:

[MY_WH]

ApplicationUsingThreads=1

AlternateServers=

Description=MicroStrategy ODBC Driver for PostgreSQL Wire Protocol

FailoverPreconnect=0

FetchTWFSasTime=1

Driver=/opt/mstr/MicroStrategy/install/lib/MYpsqlXX.so

ConnectionRetryDelay=3

EnableDescribeParam=1

InitializationString=

FetchRefCursors=1

Database=tutorial_wh

QueryTimeout=0

HostName=###.###.###.###

FetchTSWTZasTimestamp=1

ReportCodePageConversionErrors=0

FailoverMode=0

LoadBalancing=0

ConnectionRetryCount=0

ExtendedColumnMetaData=0

XMLDescribeType=-10

LoginTimeout=15

TransactionErrorBehavior=0

FailoverGranularity=0

PortNumber=5432

LogonID=mstr

Replace the IP address (###.###.###.###) in the syntax above with the Intelligence Server IP address in your own cloud environment.

15 **Save** and close the **odbc.ini** file.

You created a data source connection to your data warehouse.

Now that you have created a Data Source Name (DSN), you are ready to test the connection and make sure that data can be retrieved from the PostgreSQL database. To test the connection, import data to MicroStrategy Web.

Import data from the PostgreSQL database

1 From the Welcome to MicroStrategy Cloud email, click **Access MicroStrategy Platform**.

Open MicroStrategy Web

2 On the Login page, type the credentials from the Cloud email. The Cloud landing page opens.

3 Hover over **MicroStrategy Web** and click **Launch**.

4 Click **Go to MicroStrategy Web**. The Shared Reports page opens.

Import data

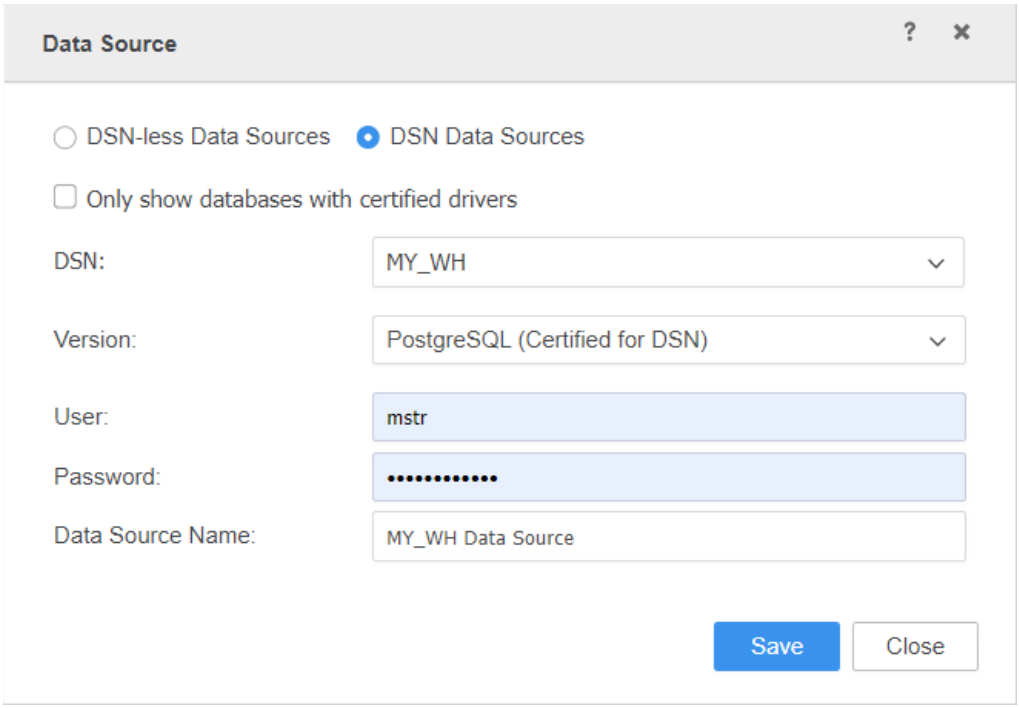
5 Click **Create** and select **Add External Data**.

6 Click **Databases**.

7 Click **Select Tables**, and then click **Next**. The Import from Tables window opens.

Specify a new data source

8 Click **New Data Source**. The Data Source window opens.



9 Click **DSN Data Sources**.

10 From the **DSN** drop-down list, select **MY_WH**. This is the DSN you added to the odbc.ini file.

11 From the **Version** drop-down list, select **PostgreSQL**.

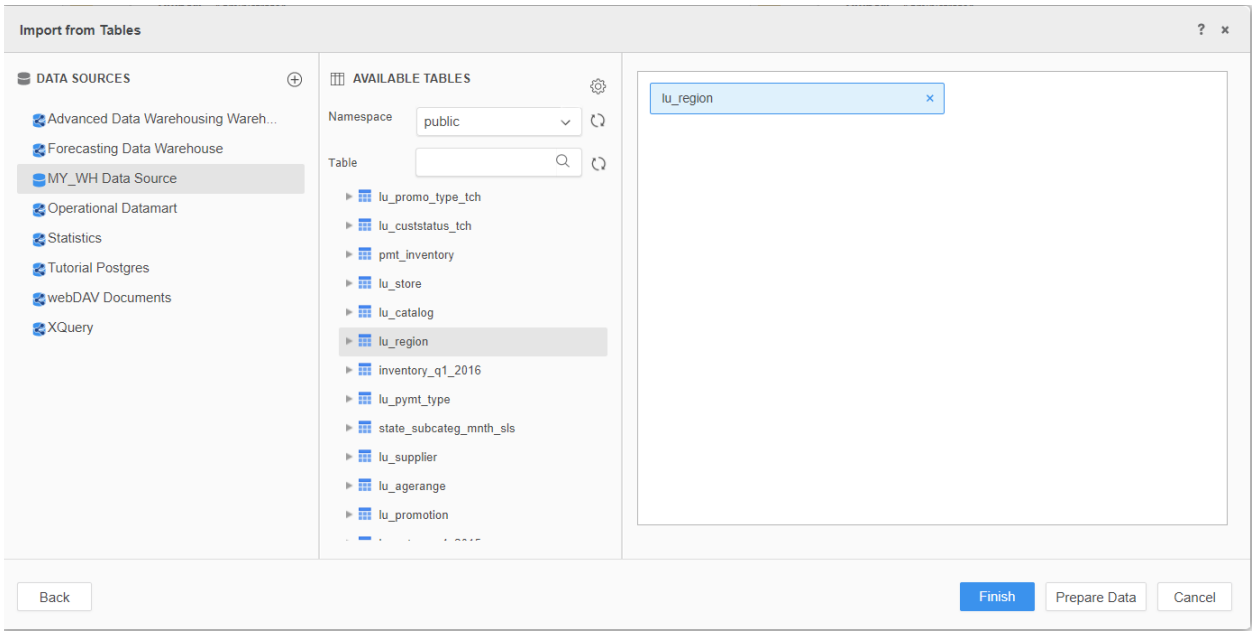
12 In the **User** and **Password** boxes, type the credentials from your Cloud email. This is the database user login.

13 In the **Data Source Name** box, type **MY_WH Data Source**. This is the Data Source display name in the Import from Tables window.

14 Click **Save**.

Import a table

15 In the Import from Tables window, under Data Sources, click **MY_WH Data Source**.



16 From the **Namespace** drop-down list, select **public**.

17 Double-click **lu_region** and click **Finish**.

18 Click **Import as In-Memory Dataset**.

19 **Save** the imported data in the **My Reports** folder.

20 Click **Create Dossier**. The attributes and metrics are displayed in the Datasets panel.

You successfully imported data from the data source you specified in the odbc.ini.

4

Securing the Infrastructure

Security is a crucial area of concern when creating and maintaining your MicroStrategy infrastructure. A secure system mitigates the loss of resources, privacy, and competitive advantage. To secure your infrastructure, establish guidelines to protect from:

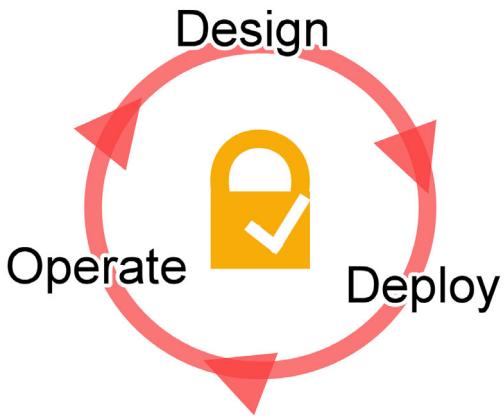
- **Attacks**: Malicious intrusions that disable your operations or expose sensitive data. Regular attacks occur in any system that is connected to the Internet.
- **Security lapses**: Procedural mistakes that produce a negative outcome. For example, an accidental exposure of private data to the Internet, or the accidental deletion of a table in your warehouse.
- **Aberrations**: An unlucky mishap that compromises your system. For example, you might experience an outage while running a data update script.

Identify possible security concerns that fall into these categories and create security policies in your organization to limit your exposure to them. In this chapter, explore the following security topics:

- Infrastructure security policies
- Secure data transfer
- System availability

Creating infrastructure security policies

Although all Intelligence Center architects play a role in securing enterprise resources, your goal is to address security concerns related to the infrastructure components that house the MicroStrategy platform. For example, you might create guidelines that ensure infrastructure security as your organization's implementation teams participate in the following workflows:



- **Infrastructure design**: Your network and hardware topology. For example, create signed security certificates and install them on the servers that house MicroStrategy components like Intelligence Server and MicroStrategy Web.
- **Infrastructure deployment**: The process and protocols used to install and configure your systems. For example, close any ports that are not necessary to prevent intruders from gaining access to your system.
- **Secure system operation**: The daily functions required to run your systems. For example, change the default administrator passwords on your web application server.

To develop your infrastructure security protocols, implement the following principles in relation to your hardware and software components.

Best Practice

Assign minimal privileges to each user

To restrict access to protected resources, limit user privileges for administrators on servers and end users on their client machines. Each user should only be assigned the level of privilege and system access that is required to perform their job functions.

For example, all administrators should not have root access to Linux servers.

Best Practice

Validate all user and client connections

Malicious code or users can be inserted into any communication chain. To mitigate attacks, each user must be assigned personal credentials to access systems and applications.

For example, use a trusted authentication provider to verify user and client identity and ensure that users have their own credentials for servers and databases.

Best Practice

Create an audit trail

Document system activity to create monitoring points and identify security failures. This process simplifies troubleshooting workflows and provides information that helps you fine-tune your systems.

For example, configure system logs that capture Intelligence Server crashes, ODBC traces, system errors, and so on.

Best Practice

Review and update default settings

Default ports and passwords in your system applications are vulnerable to rudimentary automated attacks. Default settings that are not updated expose your system to basic attacks.

For example, your web application server likely uses a default administrative password that must be updated.

Best Practice

Establish security on multiple levels

To deter malicious attack, ensure that each level of your infrastructure contains its own security layer that is not reliant on other layers. As your system evolves, monitor the infrastructure for attacks and continuously reinforce the most vulnerable security risk.

For example, establish access control in your application interfaces, services, databases, and networks.

An obscure system is not necessarily secure

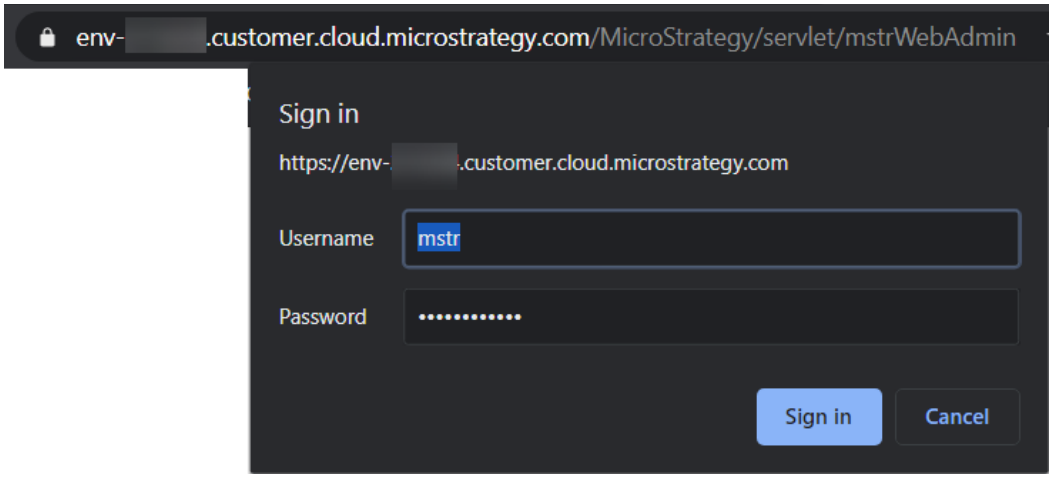
Simply hiding various facets of your infrastructure does not promote security. Assume that malicious actors are already familiar with your system topology. Embrace a simple security model that leverages proven security solutions.

As you design and implement your infrastructure, keep these fundamental security principles in mind to reduce vulnerabilities and mitigate risk.

Exercise 4.1: Apply security principles to Tomcat users

The Platform Administrator wants to assign MicroStrategy Web administrative duties to a specific person on the administration team, and needs you to create the user on the Tomcat web application server. Based on the security principles you have established at InfiniRec, you want to assign the least number of privileges required for the user to perform their work.

In this exercise, create a new Tomcat user that is only able to access and modify the MicroStrategy Web Administrator page, as displayed in the following image.



To create the new user with the required privileges, assign the admin and mstrWebAdmin roles that have already been configured in Tomcat.

Create a new Tomcat user

Connect to the Linux server

1. From the Cloud landing page, hover over Remote Desktop Gateway and click **Launch**.
2. If you see the See Text and Images Copied to Clipboard pop-up window, click **Allow**.
3. Log in with the credentials from your Welcome to MicroStrategy Cloud email.

4 In the All Connections area, click **Platform Instance SSH**. A secure shell to the Linux server is displayed.

Back up the Tomcat user configuration file

5 Navigate to the Tomcat configuration folder. To do this, type:

```
cd /opt/apache/tomcat/apache-tomcat-9.0.30/conf
```

6 The Tomcat passwords have already been configured in your environment. To avoid accidental modifications to your environment, create a copy of the configuration file for experimental purposes. To do this, type:

```
cp tomcat-users.xml tomcat-users-backup.xml
```

7 To verify that the tomcat-users-backup.xml file was created, type:

```
ls
```

Make sure that tomcat-users.xml and tomcat-users-backup.xml are both listed.

Edit the Tomcat users

Now that you have a backup of your configuration file, you can make changes to the original file.

8 To open the tomcat-users.xml file, type:

```
nano tomcat-users.xml
```

9 Press **Ctrl + V** to go to the next page.

The default Tomcat users are commented. These users are inactive:

```
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
```

Aside from the default Tomcat roles and users, four roles (admin, manager, cloud, and mstrWebAdmin) were created and assigned to the mstr user:

```
<role rolename="admin" />
<role rolename="manager" />
<role rolename="cloud" />
<user password="OJnKwq1uYWit" roles="admin,manager,cloud,mstrWebAdmin" username="mstr" />

<role rolename="mstrWebAdmin" /></tomcat-users>
```

10 To create a new user, position the cursor on an empty line and type the following:

```
<user username="myWebAdmin" password="So$3cure"
roles="admin,mstrWebAdmin" />
```

Do not copy and paste.

Your file looks similar to the following:

```
<role rolename="admin" />
<role rolename="manager" />
<role rolename="cloud" />
<user password="SL4iusr5hx4P" roles="admin,manager,cloud,mstrWebAdmin" username="mstr" />
<user username="myWebAdmin" password="So$3cure" roles="admin,mstrWebAdmin" />
<role rolename="mstrWebAdmin" /></tomcat-users>
```

11 Press **Ctrl** + **X** to exit.

12 At the Save prompt, type **Y** to save your changes.

13 At the File Name prompt, press **Enter** to keep the existing file name.

Test the myWebAdmin user

14 Clear your browser cache. To do this in Chrome:

a Click the menu in the top-corner.

b Point to **More Tools**.

c Click **Clear Browsing Data**.

d From the **Time Range** drop-down list, select **Last Hour**.

e Click **Clear Data**.

15 In your browser, navigate to

https://env-**XXXXXX**.customer.cloud.microstrategy.com/MicroStrategy/servlet/mstrWebAdmin

where **XXXXXX** is your environment number

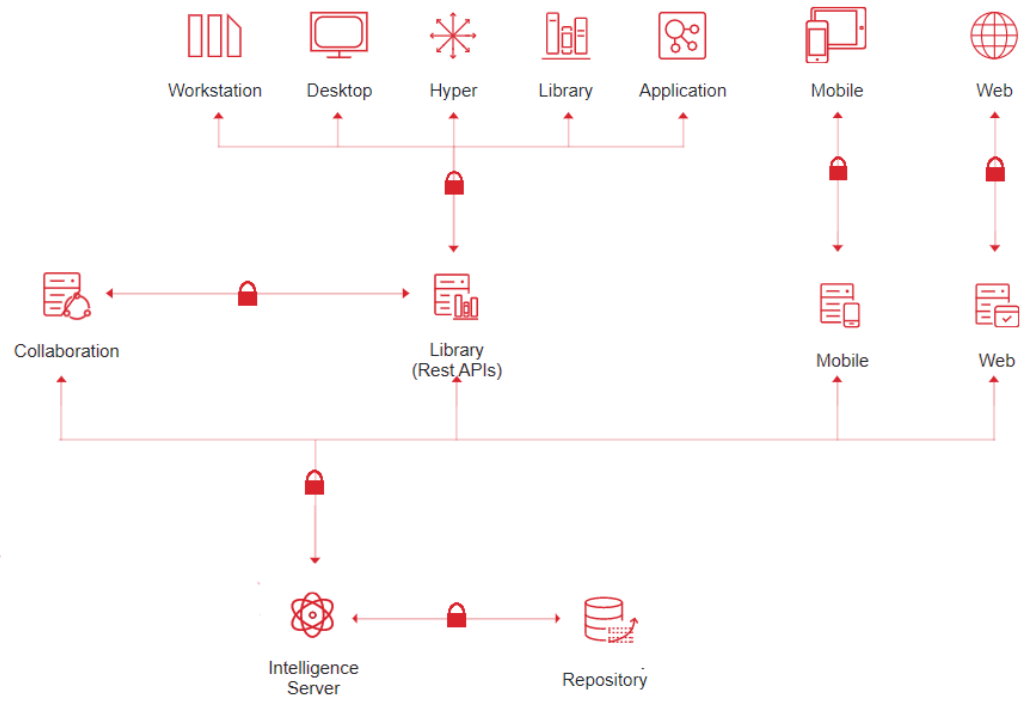
16 In the Sign In window, type the user credentials you created. The MicroStrategy Web Administrator page is displayed.

You created a new Tomcat web application user based on the established security principles at InfiniRec.

Establishing secure data transfer practices

Analyzing your organization's data can help stakeholders improve workflows, foster relationships with customers, and create a competitive advantage. Because data is critical to your operation, you must secure it as it is transferred through

internal networks and the Internet. For example, you can secure the connections between machines that host the following MicroStrategy components:



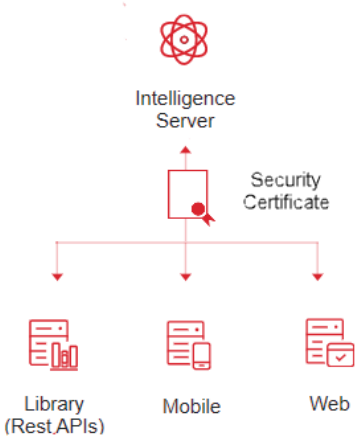
- Intelligence Server machines and application servers such as MicroStrategy Web and Mobile
- Intelligence Server machines and data sources
- MicroStrategy Library and Collaboration Server
- Your web application server and client browsers

The Platform Administrator can configure the Intelligence Server to transfer data without any encryption. However, if your organization's data is private or sensitive, it must be encrypted before it is transferred from the Intelligence Server to application servers that host services like MicroStrategy Web and Mobile. Work with the Platform Administrator to establish secure data transfer policies for MicroStrategy services and the machines that host those services.

Verifying secure communication with certificates

Secure data transfer over a network can be established through Secure Socket Layer (SSL) or Transport Layer Security (TLS) encryption. For example, Intelligence Server uses port 39320 to communicate securely with other servers through SSL.

Once your servers have been configured to use encryption technologies, you must install signed security certificates that validate your security configuration to client machines. For example, Library, Mobile, and Web validate the security certificate installed on your Intelligence Server machine before they establish a connection.



As an added layer of protection, the Platform Administrator can also require application servers to be validated before communication is established with the Intelligence Server. To comply with this policy, a security certificate must be installed on each application server and added to the Intelligence Server's truststore.

As the System Administrator, your role is to develop standards for the creation, conversion, and installation of certificate files, and work with the Platform Administrator to identify the machines that require signed certificates.

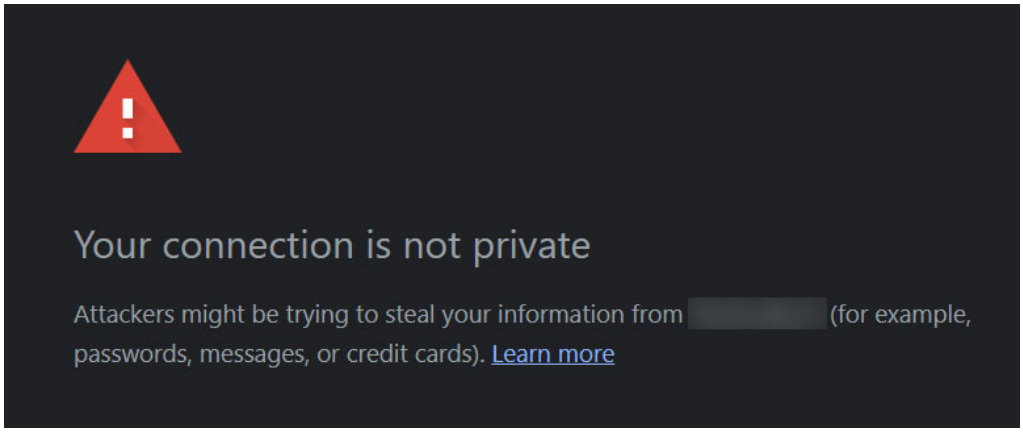
Securing communication between the web server and client browsers

In many MicroStrategy environments, analysts create and execute reporting objects through the MicroStrategy Web application. To facilitate this connection, a web application server hosts a web site that users connect to through their web browsers.

You can configure your web application server to encrypt data transferred to clients. A secure connection requires users to access MicroStrategy Web through their browsers using HTTPS, as in the URL you use to connect to your Cloud environment's Web instance:

https://env-XXXXXX.customer.cloud.microstrategy.com//MicroStrategy/servlet/mstrWeb

If your MicroStrategy Web web site does not have a valid security certificate installed, the following message is displayed in the browser when users access the site.



Browsers display a security warning when they encounter a variety of security certificate validation issues, including:

- The browser does not recognize the Certificate Authority (CA) signature
- The certificate is expired
- The certificate is for a different site
- A certificate is not installed

To establish a secure connection between the web application server and client browsers, you must:

- Configure the Tomcat server to use SSL encryption
- Create a private key and Certificate Signing Request (CSR) for your server
- Submit your CSR to a Certificate Authority (CA)

Once you receive the signed certificate from the CA, install it on your Tomcat server along with the CA's root certificate and your server's private key.

Exercise 4.2: Explore the Tomcat server's security configuration

In your Cloud environment, the Tomcat web application server installed on your Linux server hosts the MicroStrategy Web web site. The communication between your browser and the Tomcat server is encrypted using SSL.

In this exercise, examine the Tomcat server's security certificate and SSL configuration.

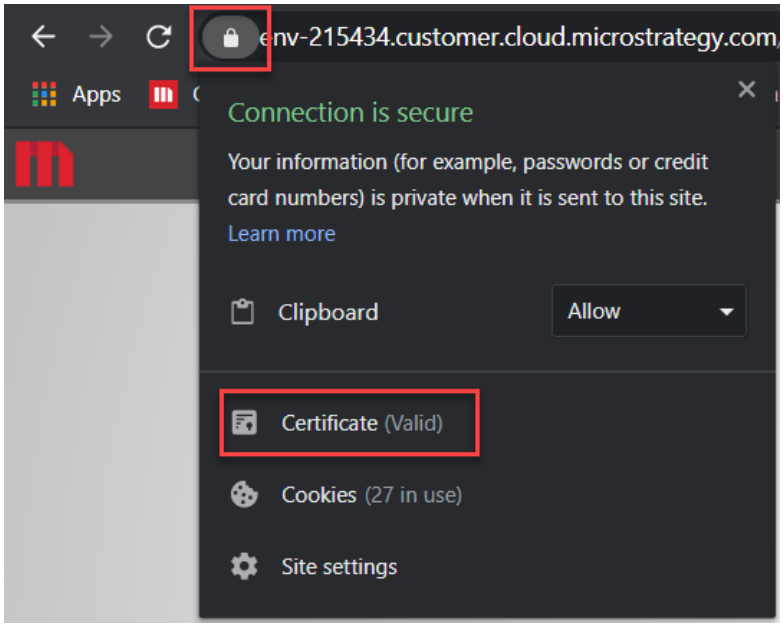
View the security certificate

1 In Chrome, navigate to:

https://env-*XXXXXX*.customer.cloud.microstrategy.com/MicroStrategy/servlet/mstrWeb

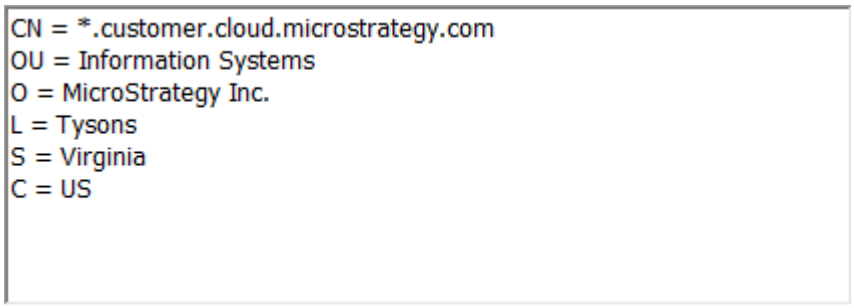
where *XXXXXX* is your environment number.

2 In the address bar, click **View Site Information**. The Connection is Secure window opens.



3 Click **Certificate**. The Certificate window opens and displays the certificate details.

4 Click the **Details** tab, and then click **Subject**. The following information is displayed:



- **CN**: Common Name for the server
- **OU**: Organizational Unit that is responsible for installing and maintaining the certificate
- **O**: Organization to which the certificate is issued
- **L**: The organization's locality
- **S**: The organization's state

You viewed the certificate installed on the Tomcat server. When analysts access MicroStrategy Web, their browsers validate the *.customer.cloud.microstrategy.com domain's identity by verifying the security certificate details.

Now that you have examined the security certificate, investigate the Tomcat server's security configuration.

Examine the Tomcat security configuration

1 From the Cloud landing page, open the Apache Guacamole SSH connection to your Linux server.

2 Navigate to the Tomcat configuration folder:

```
cd /opt/apache/tomcat/latest/conf
```

3 Open the Tomcat server configuration file:

```
nano server.xml
```

4 Scroll down to the line that begins with <Connector SSLEnabled="true"

5 Identify the `keystore` file value. This is the storage location for the keystore file, which contains the MicroStrategy Web web site's private key, signed certificate, and the CA's root certificate.

6 Identify the `keystorePass` value. This is the password used to secure the keystore file.

7 Press **Ctrl** + **X** to exit the file without saving changes.

8 View the contents of the keystore file. To do this, type:

```
keytool -list -v -keystore /opt/apache/tomcat/
latest/keystore -storepass opsworkstomcat
```

The installed certificate is displayed as a `privatekeyentry`. which contains your server's signed certificate and private key.

```
Alias name: 1
Creation date: Jun 8, 2020
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: OU=Cloud, O=MicroStrategy, L=Tysons Corner, ST=VA, C=US
Issuer: OU=Cloud, O=MicroStrategy, L=Tysons Corner, ST=VA, C=US
Serial number: b9938f6fcd88f5ba
Valid from: Fri Oct 14 13:44:38 UTC 2016 until: Mon Oct 12 13:44:38 UTC 2026
Certificate fingerprints:
         MD5:  EF:96:DD:19:F9:CE:F5:63:AE:B3:61:77:4B:86:48:78
         SHA1: 4A:F6:58:20:4A:8C:62:3A:CD:82:03:FC:3B:D6:E4:8F:81:F8:77:DB
         SHA256: 25:76:93:28:07:F1:1F:03:32:D6:36:12:BE:C9:6B:7D:0D:6F:10:A9:FA:BA:64:AA:6E:A8:55:28:0C:B9:BF:F2
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3
```

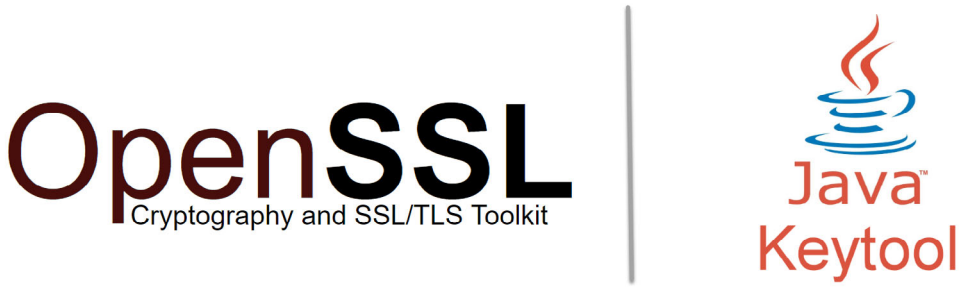
You examined the Tomcat server's SSL configuration as well as the keystore file that holds the certificate files.

Managing security certificates

To establish secure communications in your MicroStrategy environment, you must perform the following security certificate management tasks:

- Create private key and certificate files for your servers
- Create Certificate Signing Requests (CSRs) to submit your certificates for signatures by a third-party
- Sign your own certificates for non-production environments
- Combine private key and certificate files into a keystore file

To perform these tasks, use OpenSSL and the Java Keytool.



These open source command line tools are installed by default on the Linux and Windows servers in your MicroStrategy Cloud environment. In your own organization, download the desired version of these tools based on the operating systems installed on your on-premise infrastructure.

Constructing OpenSSL commands

OpenSSL commands are constructed with an initial command and optional parameters to specify the command's details For example, the following command generates a private key for the machine:

```
openssl genrsa -des3 -out filename
```

where:

- `openSSL` initiates the OpenSSL tool
- `genrsa` is a command that generates a private key using an RSA algorithm
- `-des3` is an option to use the Triple Data Encryption Algorithm to encrypt the file
- `-out` *`filename`* is an option that outputs the private key to the specified file

Executing keytool commands

The Java keytool commands are constructed using a syntax similar to that of OpenSSL. For example, the following command can be used to import one keystore file into another:

```
keytool -importkeystore -srckeystore filename
-destkeystore filename -srcstoretype filetype -alias
aliasname
```

where:

- `keytool` initiates the Java keytool
- `-importkeystore` is a command that imports a source keystore file into a destination keystore file
- `-srckeystore` *`filename`* is an option that specifies the source keystore filename, for example, `-srckeystore sourcefile.p12`
- `-destkeystore` *`filename`* is an option that specifies the destination keystore filename, for example, `-destkeystore destinationkeystore.p12`
- `-srcstoretype` *`filetype`* is an option that specifies the source keystore file type, for example, `-srckeystore sourcefile.p12`
- `-alias` *`aliasname`* is an option that specifies the source keystore alias name that is used to identify the source keystore entry in the destination keystore file, for example, `-aliasname tomcat`

Both OpenSSL and Java keytool commands can be used to create and convert certificate files. The keytool can also be used to manage keystore files that store key information along with other certificate files. Use the syntax outlined above along with the documentation for each tool to manage certificate files in your own environment.

Exercise 4.3: Create a Certificate Signing Request (CSR)

The certificate installed on the Tomcat web server in your Cloud environment was signed by a third-party Certificate Authority (CA). To receive a signature from a CA, you must submit a Certificate Signing Request (CSR).

At InfiniRec, MicroStrategy Web is accessed through a web site hosted on your Tomcat web application server. The site uses the SSL protocol to ensure secure data transfer between the server and client browsers. You want to install a security certificate that validates the InfiniRec website to client browsers. To do this, create a Certificate Signing Request (CSR) from the server that hosts the Tomcat application hosting MicroStrategy Web. The CSR can then be sent to an authorized CA for a signature.

Create a CSR that includes:

- The server's common name. For example, www.infinirec.com
- Your organization's name and location. For example InfiniRec in Tysons, VA
- The server's private key type. For example, RSA
- The server's key size. For example, 2048 bit

The following example displays the contents of a CSR file.

```
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=US, ST=Virginia, L=Tysons, O=InfiniRec, OU=System Administration, CN=www.infinirec.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
```

In this exercise, create a private key for your Linux server and then create a CSR that you can submit to a third-party CA.

Create a CSR

1 Open the Apache Guacamole SSH connection to the Linux server.

2 Navigate to the Documents folder:

```
cd Documents
```

3 Create a new folder to hold certificate files:

```
mkdir certificate
```

4 Navigate to the certificate folder:

```
cd certificate
```

5 Create a private key for your Linux server:

```
openssl genrsa -des3 -out myLinuxServer.key
```

6 At the **Enter pass phrase** prompt, type **L!nux$ecur3**.

7 Enter the pass phrase again.

8 List the contents of the certificates directory to verify that the key was created:

```
ls
```

The myLinuxServer.key file is listed.

Create a Certificate Signing Request

9 Create a CSR using the key file you just generated:

```
openssl req -new -key myLinuxServer.key -out
myLinuxServer.csr
```

At the prompts, type the following answers:

a Enter pass phrase for myLinuxServer.key: **L!nux$ecur3**

b Country Name (2 letter code): **US**

c State or Province Name: **Virginia**

d Locality Name: **Tysons**

e Organization Name: **InfiniRec**

f Organizational Unit Name: **System Administration**

g Common Name: **www.infinirec.com**

h Email: *your email address*

i A challenge password: **challenging123**

j An optional company name: **InfiniRec**

10 List the contents of the certificates directory to verify that the CSR was created:

```
ls
```

The myLinuxServer.csr file is listed.

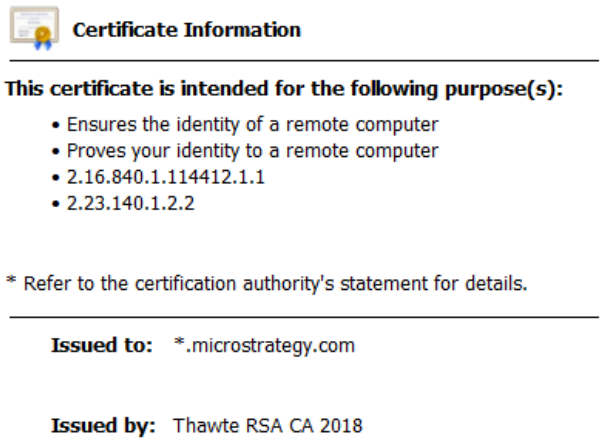
11 To verify the contents of the CSR file, type:

```
openssl req -in myLinuxServer.csr -noout -text
```

You created a CSR that includes your Linux server's private key and other identifying information about the server. To validate your server's identity to client machines, you must submit your CSR to a third-party Certificate Authority (CA) for a signature. The CA returns a signed certificate, along with a CA root certificate, which you must install on your Linux server.

Signing certificates with your Certificate Authority

In a production environment, the security certificates installed on your machines must be signed by a trusted third-party Certificate Authority (CA), which completes a validation process to ensure that you are the owner of the domain being certified. The following example certificate is signed by Thawte.



Client applications such as web browsers are configured to trust the signature of a specific list of CAs. When a client application connects to your server, it reads the certificate signature to verify the server's identity and ensure that the connection is secure.

In some cases, your team might create environments for development or proof-of-concept purposes. In these non-production environments, you can set up your own CA to sign certificates. Because client applications do not recognize your CA's signature, you must add your CA's root certificate to client truststores to complete the setup process.

Do not use self-signed certificates in a production environment.

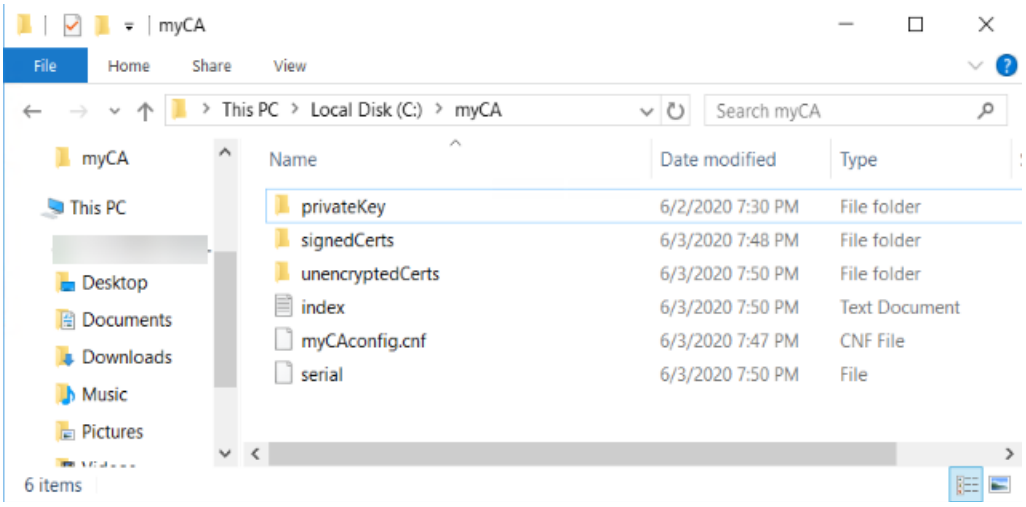
Exercise 4.4: Create your own Certificate Authority

You have been asked to create a proof-of-concept MicroStrategy environment for InfiniRec's workout apparel subsidiary, GoGoGrapefruit. Because this is a non-production environment, you decide to set up your own development CA to sign security certificates.

In this exercise, create a folder structure to store the CA's:

- private key
- signed certificates
- certificate tracking files
- configuration files

Your completed CA folder structure resembles the following example:



Create the CA folder structure

1 In File Explorer, navigate to **C:**.

2 Create a root folder for your development certificate authority. To do this:

   a Right-click an empty area, point to **New** and click **Folder**.

   b Name the folder **myCA**.

   c Double-click **myCA** to open it.

3 In the **C:\myCA** folder, create the following subfolders:

- **privateKey**
- **signedCerts**
- **unencryptedCerts**

4 In the **C:\myCA** folder, create a file to track the serial number for the next certificate created using OpenSSL. To do this:

a From the taskbar, search for and open **Notepad**.

b In Notepad, type **01**.

c From the **File** menu, click **Save As**.

d Navigate to **C:\myCA**.

e In the **File Name** box, type **"serial"**.

Do not omit the quotes.

f From the **Save As Type** drop-down list, select **All Files**.

g Click **Save**.

5 In **C:\myCA**, create a file that serves as a database for issued certificates. To do this:

a From the Notepad **File** menu, click **New**.

b From the **File** menu, click **Save As**.

c Navigate to **C:\myCA**.

d In the **File Name** box, type **index**.

e From the **Save As Type** drop-down list, select **Text Documents (*.txt)**.

f Click **Save**.

g Close **Notepad**

6 In File Explorer, navigate to the OpenSSL configuration folder, **C:\Program Files (x86)\Common Files\MicroStrategy\Apache\Apache24\conf**.

7 Copy **openssl.cnf** and paste it in **C:\myCA**.

8 Rename **openssl.cnf** as **myCAconfig.cnf**.

The folder structure for your CA is complete.

Create a private key and root certificate

Create a private key and root certificate that identify your CA on this Windows server.

1 Open a command prompt. To do this, from the taskbar, search for **cmd**.

2 Navigate to the OpenSSL installation folder. To do this, type:

```
cd C:\Program Files (x86)\Common Files\
MicroStrategy\Apache\Apache24\bin
```

3 Create a private key and a root certificate. To do this, in the cmd prompt, type:

```
openssl req -config C:\myCA\myCAconfig.cnf -new
-x509 -extensions v3_ca -keyout C:\myCA\privateKey\
myCAprivatekey.key -out C:\myCA\signedCerts\
myCArootcertificate.crt -days 365
```

Type the following information when prompted:

a Enter PEM pass phrase: **my$ecur3CA**

b Enter the pass phrase again

c Country Name: **US**

d State or Province Name: **Virginia**

e Locality Name: **Tysons**

f Organization: **My Certificate Authority**

g Organizational Unit: **System Administration**

h Common Name: **env-XXXXXX.customer.cloud.microstrategy.com**

i Email Address: ***your email***

The private key and root certificate for your CA are created.

4 In the **C:\myCA** folder, open the **privateKey** and **signedCerts** folders to see your artifacts.

Configure OpenSSL to use your CA's private key and root certificate

5 In **C:\myCA**, right-click **myCAconfig.cnf** and click **Open with Notepad++**.

6 Scroll to the section that begins with **[CA_default]** and make the following changes to update the location of your CA's files and folders:

a dir = **C:\\myCA**

b certs = **$dir\\signedCerts**

c database = **$dir\\index.txt**

d new_certs_dir = **$dir\\unencryptedCerts**

e certificate = **$dir\\signedCerts\\myCArootcertificate.crt**

f serial = **$dir\\serial**

g private_key = **$dir\\privateKey\\myCAprivatekey.key**

7 **Save** the file and close Notepad++.

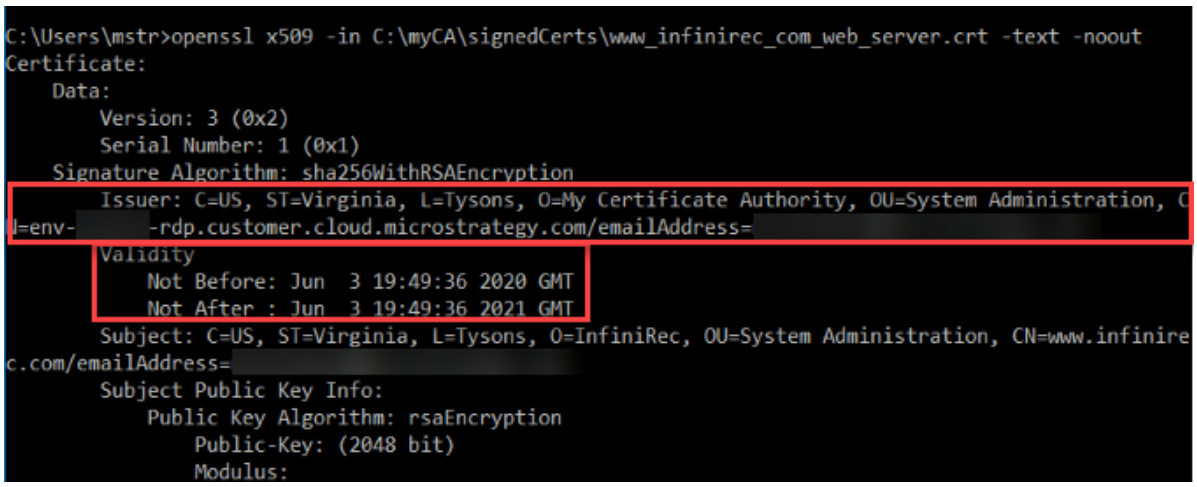
You created the CA folder structure and configuration files. You can now use your CA to sign security certificates for non-production environments.

Exercise 4.5: Sign a Certificate Signature Request

A signed security certificate installed on a server enables clients to verify the server's identity. In a production environment, certificates must be signed by a third-party CA, but in a demo or development environment, you can sign the certificates with your own CA.

In an earlier exercise, you created a Certificate Signature Request (CSR) for your Linux server. In this exercise, download the CSR to your Windows server and sign it with the GoGoGrapefruit CA.

After you sign a certificate, it will contain your CA's details and an expiration date, as in the following example.

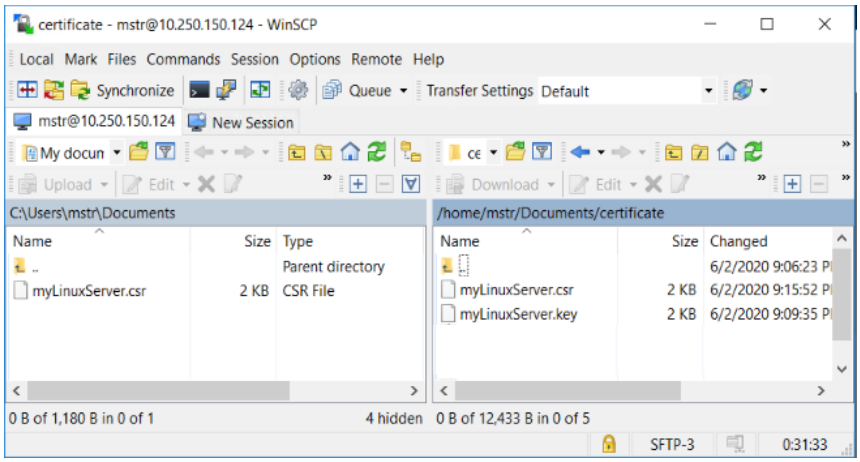


Download the CSR from your Linux server

1. From the Windows desktop, double-click **hosts** and open it with **Notepad**.
2. At the bottom of the hosts file, copy the Linux server IP address.
3. From the Windows desktop, double-click **WinSCP** and do the following in the Login window:
   a. In the **Host Name** box, paste the Linux server IP address.
   b. In the **User Name** and **Password** boxes, type the credentials from your Welcome to MicroStrategy Cloud email.
   c. Click **Save**.
   d. Select **Save Password** and then click **OK**.

e Click **Login**.

4 In the right pane, navigate to **home/mstr/Documents/certificate**.



5 In the left pane, navigate to **C:\users\mstr\Documents**.

6 Select **myLinuxServer.csr** and click **Download**. The CSR is downloaded to your Windows machine.

You downloaded CSR from the Linux server to your Windows server, where your CA is installed.

Sign the CSR using your development certificate authority, myCA

1 From the Windows taskbar, search for and open **cmd**.

2 Navigate to the OpenSSL installation folder. To do this, type:

```
cd C:\Program Files (x86)\Common Files\
MicroStrategy\Apache\Apache24\bin
```

3 Use your CA to sign the CSR. To do this, type:

```
openssl ca -config C:\myCA\myCAconfig.cnf -policy
policy_anything -out C:\myCA\signedCerts\
www_infinirec_com_web_server.crt - infiles C:\
Users\mstr\Documents\myLinuxServer.csr
```

Type the following information when prompted:

a Enter pass phrase: **my$ecur3CA**

b Sign the certificate: **y**

c 1 out of 1 certificate requests certified, commit? **y**

4 In File Explorer, open the **C:\myCA** folder and double-click **index.txt**. The newly signed certificate is listed.

5 Open the **signedCerts** folder. The **www_ininirec_com_web_server.crt** certificate is displayed.

6 To verify the certificate details, in Command Prompt, type:

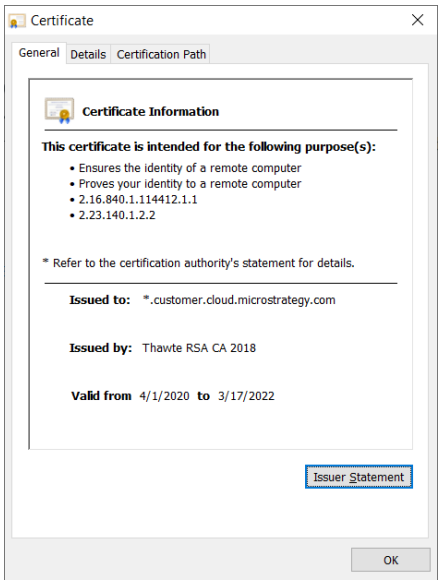
```
openssl x509 -in C:\myCA\signedCerts\
www_infinirec_com_web_server.crt -text -noout
```

You used your CA to sign a CSR generated on your Linux server. Now that you have a signed certificate, you are ready to install it on your Tomcat server to establish a secure connection between the web application server and client browsers in your non-production environment.

Exercise 4.6: Install a signed certificate on the Tomcat server

To enable secure communication between your Tomcat web server and client browsers, you must install a signed certificate along with the Certificate Authority's root certificate on your Tomcat server. This process enables client browsers to verify your web server's identity and establish a secure connection.

Your browser validates site certificates in the background and only warns you if it encounters a problem. You can view certificate details for any site you visit, as in the following example:



The Tomcat server in your Cloud environment already has a security certificate signed by a third party. This certificate is automatically trusted by major web browsers.

In contrast, the certificate you created for GoGoGrapefruit's MicroStrategy Web web site is signed by GoGoGrapefruit's own CA, which is only suitable for a non-production environment and is not trusted by common browsers.

Although the Tomcat server already has a valid security certificate, install your newly signed certificate to understand the installation process.

In this exercise, perform the following tasks:

- Upload the following files to the Linux server
  - The self-signed certificate for the Linux server

  - Your CA's root certificate
- Combine the following files into a single keystore
  - The signed certificate for the Linux server
  - The private key for the Linux server
  - Your CA's root certificate
- Import your combined keystore file into the Tomcat server's keystore

Import a new keystore into the Tomcat keystore

1. From the Windows desktop, open WinSCP.
2. In the left pane, navigate to **C:\myCA\signedCerts**.
3. In the right pane, navigate to **/home/mstr/Documents/certificate**.
4. Select the following files:
   - **myCArootcertificate.crt**
   - **www_infinirec_com_web_server.crt**
5. Click **Upload**.

Install the certificates on the Linux server:

6. Open the Guacamole SSH connection to your Linux server.
7. Navigate to the Java Development Kit installation location:

```
cd /opt/jdk/jdk8u242-b08/bin
```

8. Execute an OpenSSL command to combine your Linux server's newly signed certificate, your Linux server's private key, and your CA's root certificate:

```
openssl pkcs12 -export -in /home/mstr/Documents/
certificate/www_infinirec_com_web_server.crt -inkey
/home/mstr/Documents/certificate/myLinuxServer.key
-out /home/mstr/Documents/certificate/
myLinuxKeystore.p12 -name tomcat -CAfile /home/
mstr/Documents/certificate/myCArootcertificate.crt
-caname root
```

Type the following information when prompted:

a Enter pass phrase for key: **L!nux$ecur3**

b Enter export password: **Key$tor3**

c Verifying - Enter export password: **Key$tor3**

The myLinuxServer.p12 keystore file is generated.

9 Execute a keytool command to import the new keystore into the existing Tomcat keystore:

```
keytool -importkeystore -srckeystore /home/mstr/
Documents/certificate/myLinuxKeystore.p12
-destkeystore /opt/apache/tomcat/latest/keystore
-srcstoretype pkcs12 -alias tomcat
```

Type the following information when prompted:

a Enter destination keystore password: **opsworkstomcat**

b Enter source keystore password: **Key$tor3**

10 List the keystore entries in the Tomcat keystore:

```
keytool -list -v -keystore /opt/apache/tomcat/
latest/keystore
```

Type the following information when prompted:

keystore password: **opsworkstomcat**

The original entry is displayed, along with the entry (alias tomcat) that you just added. Because the Tomcat server already had a signed certificate from a third-party CA, your self-signed certificate will not be utilized.

Maintaining system availability

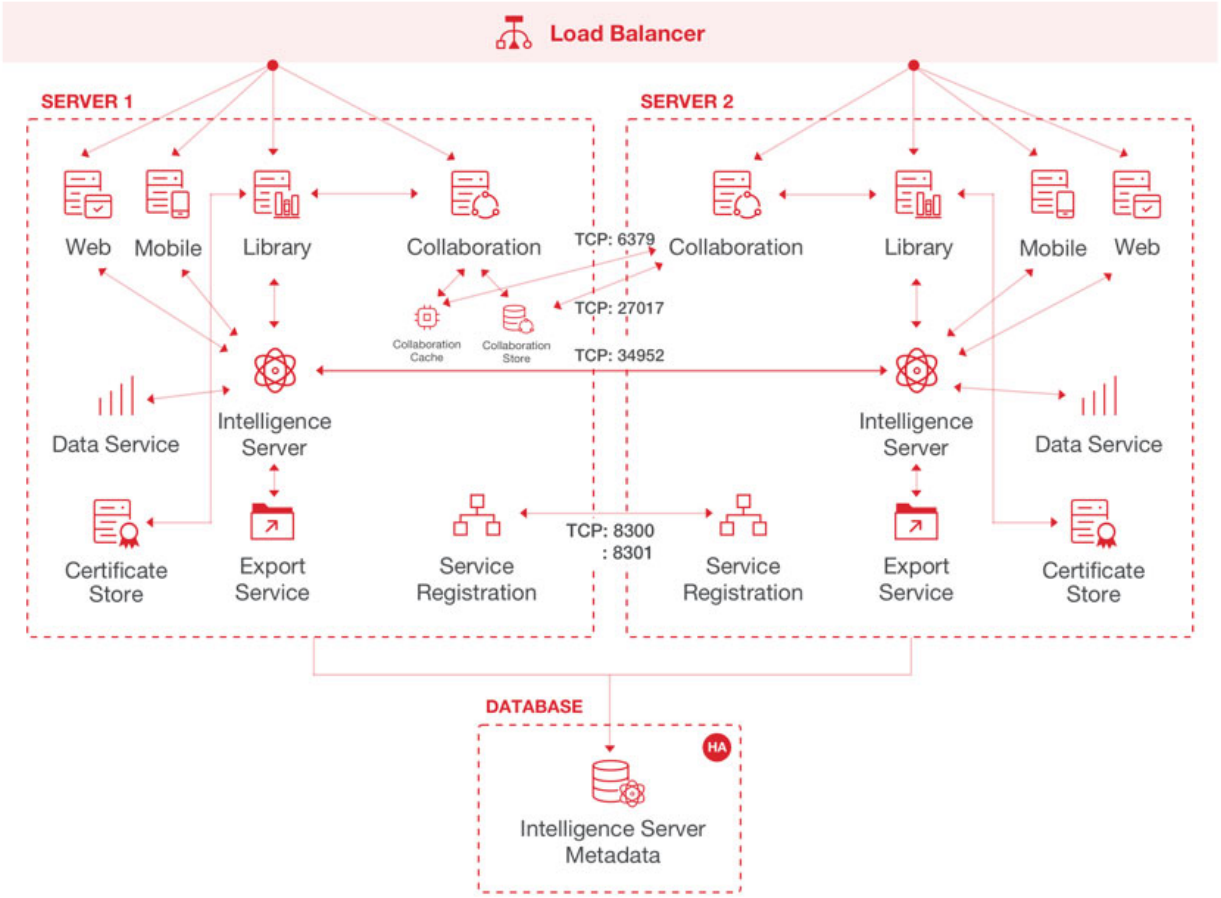
As part of your infrastructure security paradigm, ensure that the system is running and available to end users when they need to access data. MicroStrategy users leverage the platform to make critical business decisions and need to access reporting applications regardless of environmental situations, power outages, or server crashes.

To ensure that the system is always available to users, establish infrastructure guidelines that emphasize a redundant and fault tolerant system. For example, you might stipulate the following policies in your organization:

- Intelligence Servers must be installed on multiple dedicated machines
- Servers must be stored in multiple physical locations
- Security patches must be applied to server and client machines on a specific schedule

Establishing fault-tolerant environments

Infrastructure components within each environment will experience a failure of some kind during their lifespan. To mitigate the damage caused by these failures, and to ensure continuity in analytics functionality, you must develop standards to establish fault-tolerant environments. For example, you might stipulate that Intelligence Servers must be clustered with a load balancer, as in the following example.



To create a fault-tolerant environment, optimize infrastructure resources across the test, development, and production environments and create contingency plans to address failures. For example, if a component in the production environment fails, you might utilize resources from the test and development

environments as a temporary backup while you troubleshoot the problem or procure a replacement.

Create a strategy to establish a fault-tolerant architecture and provide continuity when failures occur. For example, your strategy might include the following standards:

- Cluster MicroStrategy Intelligence Servers
- Cluster database servers
- Load balance MicroStrategy Web, Mobile, and Library servers
- To ensure continuity in reporting performance, create a schedule that outlines specific times each day to back up caches, cubes, and history list files
- To ensure connectivity between MicroStrategy components and your data sources, create a backup schedule for ODBC configuration files such as odbc.ini and ODBC.sh
- To ensure continuity in MicroStrategy platform functionality, create a schedule to back up configuration, application, and customization files
- To ensure overall system continuity during emergencies such as an operating system crash or network outage, create a failover plan that enables administrators to retain reporting functionality

Creating a disaster recovery plan

There are countless circumstances that could disrupt your analytics infrastructure and prevent MicroStrategy users from accessing the information they need to make business decisions. For example, a natural disaster, extended power outage, or similar event could shut down your production environment for an extended period of time.

Because your organization's data is critical to its daily operations, you must prepare a plan that preserves data availability in the event of a catastrophic failure. For example, you might create a disaster recovery environment at a secondary sight that mirrors the production environment.

The disaster recovery plan that you create for your organization might include the following:

- A secondary location that houses a duplicate infrastructure environment.
- A process and time interval for backing up the primary production environment and duplicating it at the disaster recovery site.

- A specified cloud-based location to store your backup files that a disaster recovery team has access to.

Are you aware of your organization's disaster recovery plan and the role that you play in the plan?

5

Publishing Environment Details

Once InfiniRec's environments are up and running, you have an obligation to create documentation that helps administrators and other architects understand the infrastructure and processes required to keep it running.

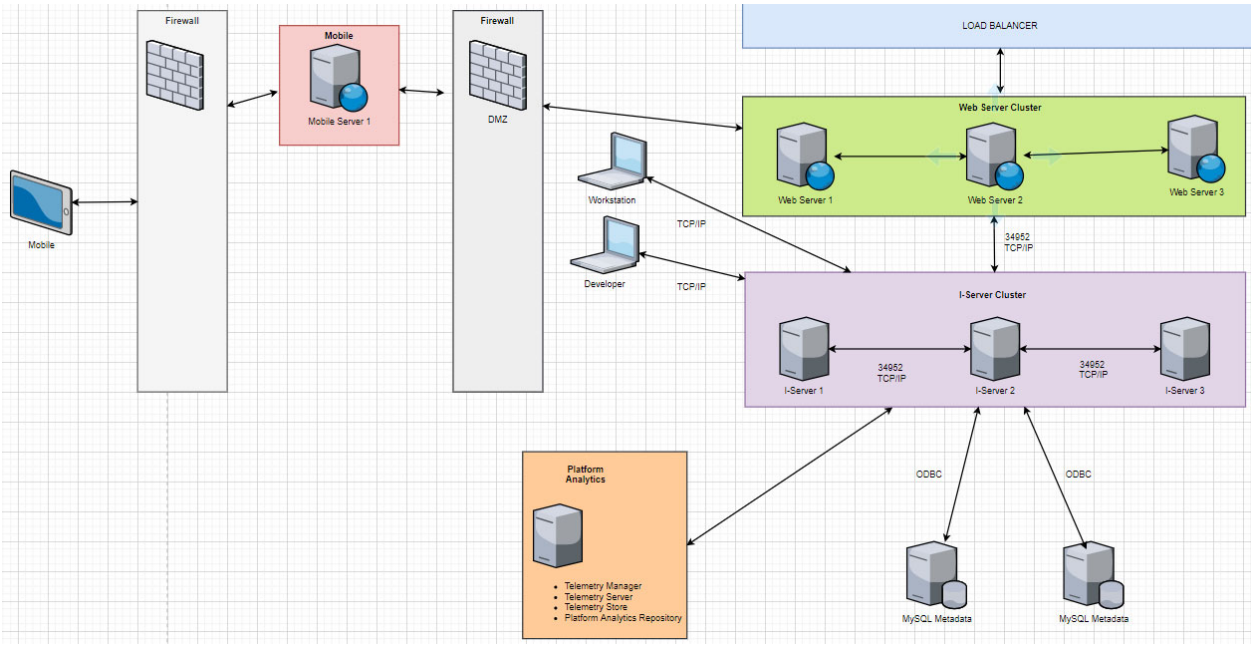
In this chapter, you will create standards to help administrators document and publish:

- **Topology diagram**: enables all parties to share a visualization of the infrastructure.
- **Standard Operating Procedures (SOPs)**: document common processes so they can be performed consistently and reliably.
- **Service Level Agreements (SLAs)**: establish responsibilities and expectations for infrastructure components and processes.
- **Server log data**: provides an audit trail for troubleshooting purposes.

The following sections discuss each of the preceding responsibilities of a System Administrator in more detail.

Creating diagrams to visualize your infrastructure

A network topology diagram is a detailed map that shows all components of each MicroStrategy platform environment, including cloud and on-premise components. The diagram details the computer network elements' positioning, and outlines the flow of data through the network. The topology diagram presents a visualization of the connections between each node in the environment and their communication channels. Components of the diagram typically include subnets, network devices like routers and firewalls, network protocols, and individual servers, as in the following example.



Creating and publishing a topology diagram helps administrators accomplish the following goals:

- Planning the MicroStrategy platform infrastructure
- Coordinating updates to an existing infrastructure
- Reporting and troubleshooting network and server problems
- Complying with security requirements
- Creating documentation for external communication, on-boarding new team members, transferring knowledge, and so on
- Tracking infrastructure components

As the System Administrator, you are responsible for creating standards that guide administrators in creating topology diagrams for each infrastructure environment in your organization. The diagram guidelines that you create might include the following standards:

- To ensure that the topology diagrams in your organization contain uniform design elements, specify the diagram creation tool that must be used by administrators. For example, you might stipulate that all diagrams must be created using Draw.io.
- Create a topology diagram for each infrastructure environment in your organization to establish continuity between your environments. For example, you may have Production, UAT, Development, and Disaster Recovery environments that each require their own diagram.
- As part of your diagram design, stipulate the minimum information that must be included in each diagram. For example, you might require diagram creators to include the following information:
  - The order of traffic flow through each component in the network.
  - Identification information for each node. For example, include each node's DNS name, IP address, operating system version, and so on.
  - Relevant network security information.
- To establish a single source for infrastructure documentation and distribute knowledge throughout the enterprise, create a central repository for topology diagrams and distribute them to the Intelligence Center.

What is your organization's preferred diagram creation tool?

Exercise 5.1: Creating a topology diagram

InfiniRec's infrastructure has been installed and configured. You want to ensure that administrators on your team and the rest of the Intelligence Center are aware of the network and infrastructure configuration so they can understand component connections, troubleshoot problems, make updates, and so on.

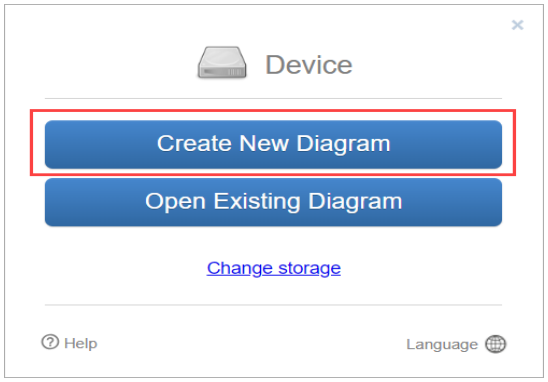
Network topology diagrams help you visualize a MicroStrategy platform environment, and aid in the process of planning infrastructure changes, troubleshooting network and server problems, sharing knowledge and so on.

In this exercise, you will create a topology diagram for a proposed MicroStrategy User Acceptance Testing (UAT) environment. As you work through the exercise, think about the diagrams infrastructure administrators will produce in your organization. Think about standards you would propose to help administrators create consistent diagrams that convey expected information about your infrastructure components.

Create a topology diagram

1 From the browser on your personal machine, open **https://draw.io**.

2 By default, the diagram will be saved to your local disk drive. Click **Create New Diagram**.

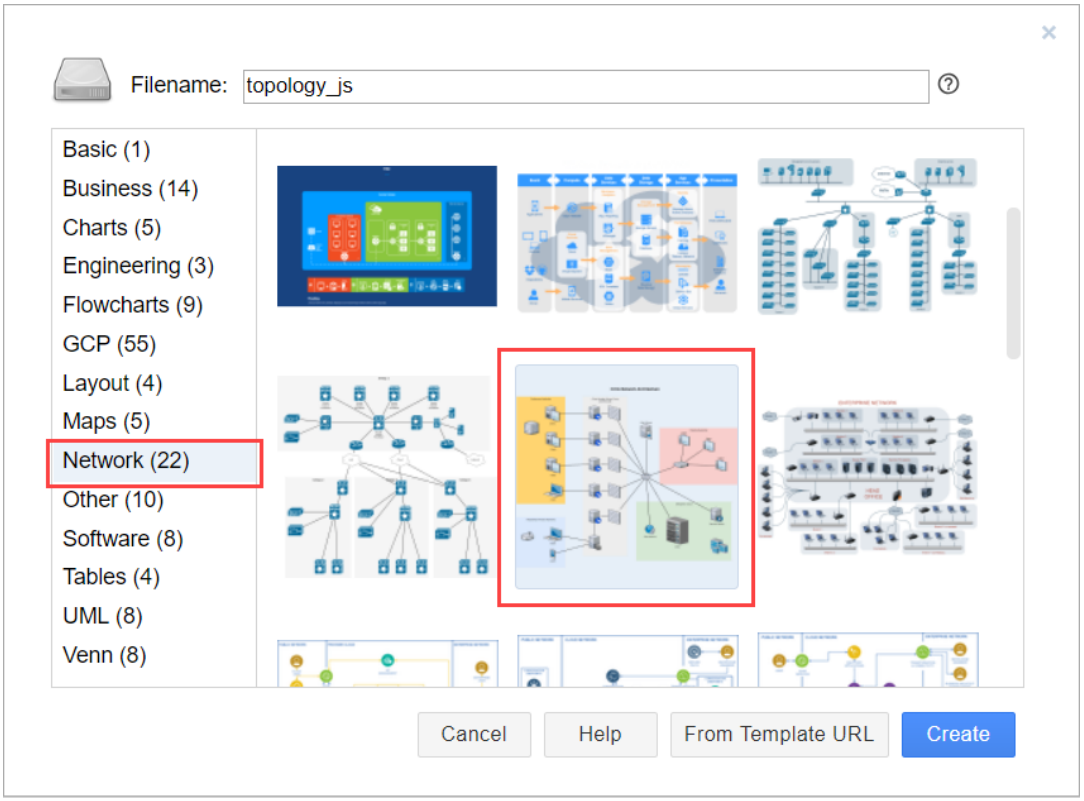


Alternatively, you can upload a template and modify it. To do this, click **Open Existing Diagram**, navigate to your **Exercise Files** folder and select **topology.drawio**.

3 In the **Filename** box, type **topology_diagram_*your initials***.

4 In the left pane, click **Network**.

5 In the right pane, click the **network/citrix** template and click **Create**. The network template opens in the editor.



6 Modify the template to add the following network components to the diagram:

- Web and Mobile users connected to the Internet
- A firewall
- Intelligence Server
- Web Server
- Library Server
- Certificate Server
- MySQL Database metadata server
- SQL Server data warehouse
- Connections between these components and the relevant port numbers

7 From the **File** menu, click **Save**. The topology diagram file is downloaded to your machine.

Documenting detailed operating instructions

Standard Operating Procedures (SOPs) document detailed instructions for regular system operations. For example, SOPs might contain a maintenance window schedule for the operating system, hardware, and other components of the infrastructure. The schedule enables the Intelligence Center to prepare for the maintenance window, and provides a trigger for administrators to notify end users.

To distribute operating awareness, create SOPs for all components of your infrastructure environments, including the operating system, network, and file system. For example, your team might create an infrastructure monitoring SOP that details the components that must be monitored, the tools used to perform monitoring tasks, the thresholds that should not be crossed, and the actions to take if the thresholds are crossed.

The SOP protocols that you develop to help infrastructure administrators create SOPs might include the following best practices:

- Ensure that task executors are involved in the creation of the SOP.
- Ensure that any administrator can easily follow and execute the SOP.
- Create a distinct SOP for each environment.

Distributing Service-Level Agreements to set infrastructure expectations

A Service-Level Agreement (SLA) is an agreement between the environment service provider and its internal or external customers. This document outlines the services the provider will furnish and defines the performance standards the

provider is obligated to meet. For example, an SLA might establish an acceptable rendering time for dashboards, as displayed in the following image.

To meet this SLA, the environment infrastructure must contain adequate hardware resources such CPU and memory.

In your organization, SLAs should document the expected quality, availability, and responsibilities for the infrastructure. For example, you can create SLAs to outline the following:

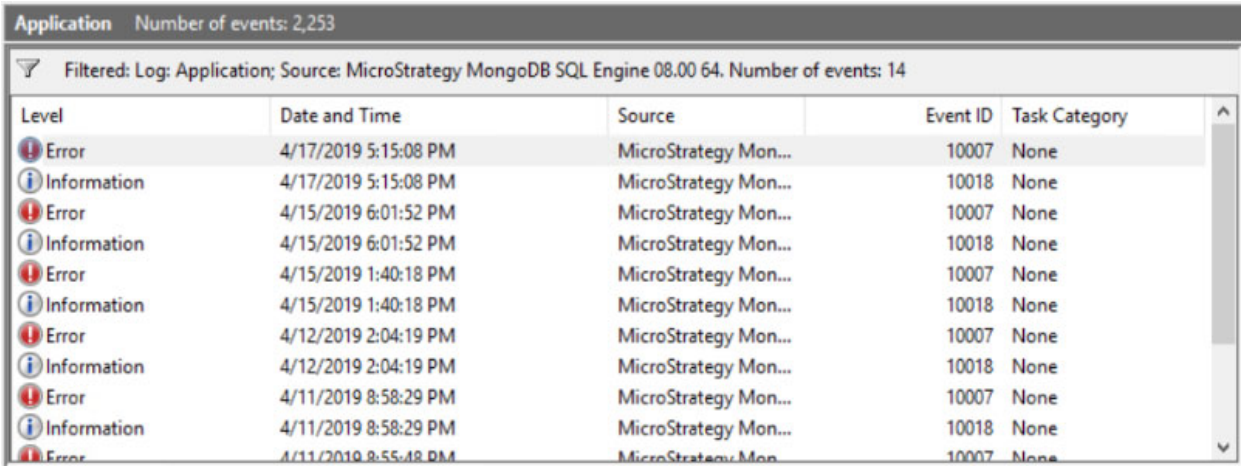
- Responsibilities of administrators in the infrastructure environments.
- The expected performance experience for end users.
- The delineation of responsibilities for infrastructure components between teams.
- The quality, availability, and responsibilities of infrastructure providers. For example, if you have a fully managed MicroStrategy Cloud environment, make the associated SLA available to the Intelligence Center and other impacted parties to establish and maintain expectations.

As part of your SLA standards, ensure that infrastructure administrators define SLAs using the SMART model - Specific, Measurable, Achievable, Relevant, and Time-bound. For example, an SLA that covers the infrastructure for a given environment might include the following points:

- All servers and MicroStrategy services will maintain an uptime of 99.9% during a calendar month. This does not include scheduled downtimes.
- OLAP cubes will be published in 60 seconds or less.
- Critical dashboards will render in 10 seconds or less.

Capturing and distributing server activity for troubleshooting

Server log files contain detailed information about activities performed on a server. These files are automatically created and maintained by each machine. For example, a server might maintain a log file that captures all operating system functions performed in a specified time period. This information can be crucial when troubleshooting a critical operating system event such as a crash. For example, the following image shows a log of errors, warnings, and information messages for a specific MicroStrategy service.



To ensure that historical server information is available for troubleshooting purposes when a critical event occurs, create a set of server activity log standards. The guidelines that you create in your organization may include the following information:

- Server activities that are critical to your infrastructure. For example, you may choose to track errors thrown by specific MicroStrategy services.
- The server machines where logging is required, as well as steps to enable and maintain server activity logs.
- Steps to enable detailed logging where desired. For example, you might include steps to log detailed network traffic information for a finite period of time when administrators need to perform troubleshooting activities.

How does your organization track server activity?

Exercise 5.2 Viewing event logs on a Microsoft Windows Server machine

The InfiniRec servers are up and running, and you want to ensure that all MicroStrategy services are functioning as expected without any errors.

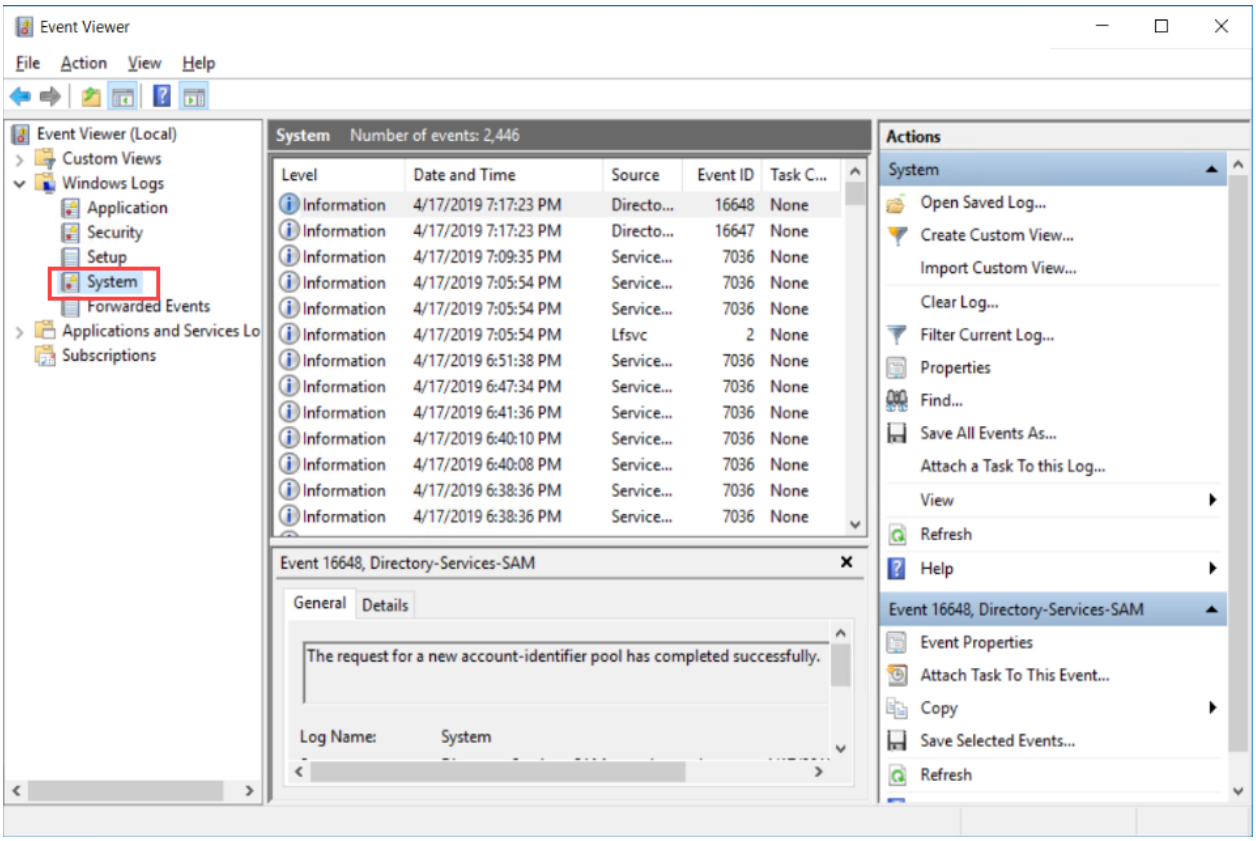
Each human interaction or running process produces an event that can be tracked in logs on your server machines. When you enable logs to capture certain events, you produce an audit trail that can be used to troubleshoot any problems that occur in your infrastructure.

In this exercise, you will explore logging options on a Microsoft Windows server.

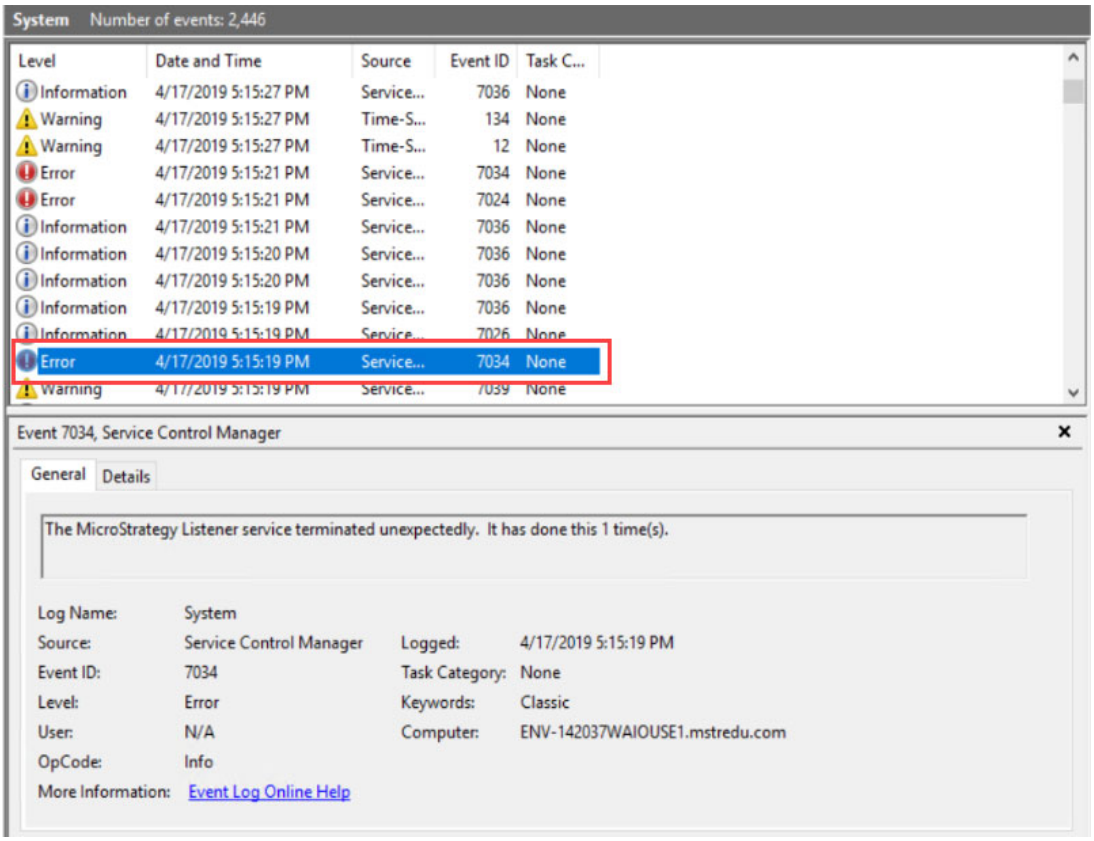
View Windows logs

1. Open the RDP connection to your Windows machine in the cloud.
2. On the task bar, in the **Search** field, type **Event Viewer**.
3. From the search results, click **Event Viewer**.

4 In the left pane, expand **Windows Logs** and click **System**. A list of events logged by the operating system is displayed in the middle pane.

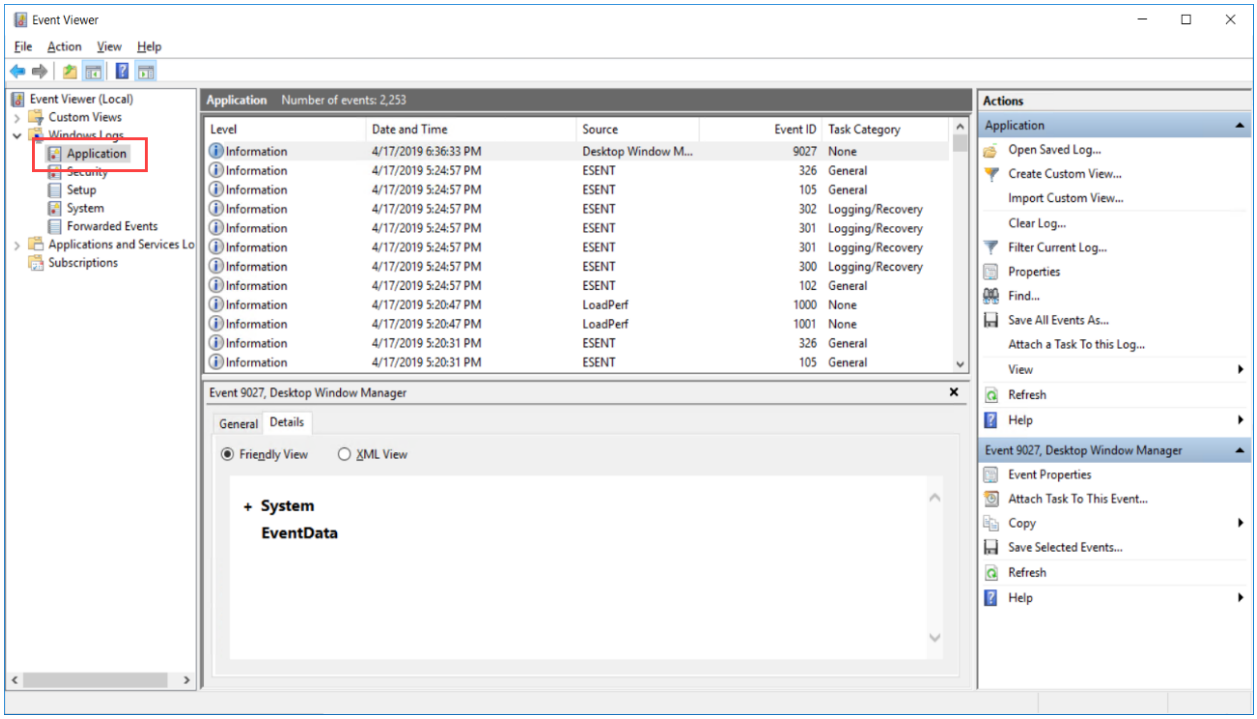


5 Click one of the events. Additional data about the event is displayed in the bottom pane.

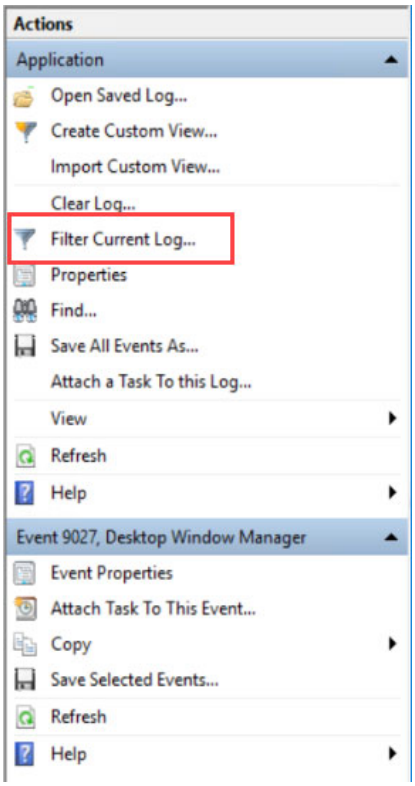


6 Review the information in the bottom pane. You can view additional details by clicking the **Details** tab.

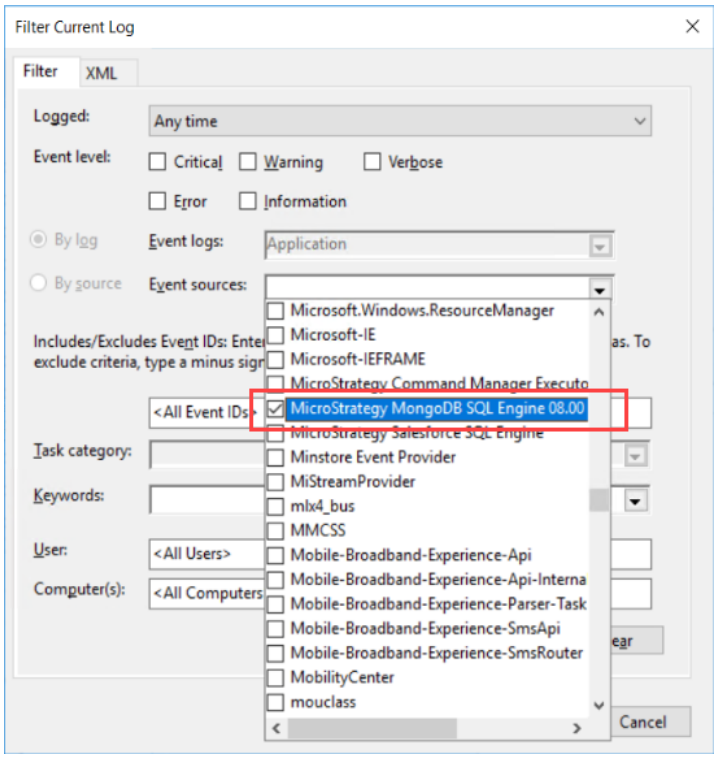
7 In the left pane, expand **Windows Logs** and click **Application**. A list of events generated by all applications on this machine is displayed in the middle pane.



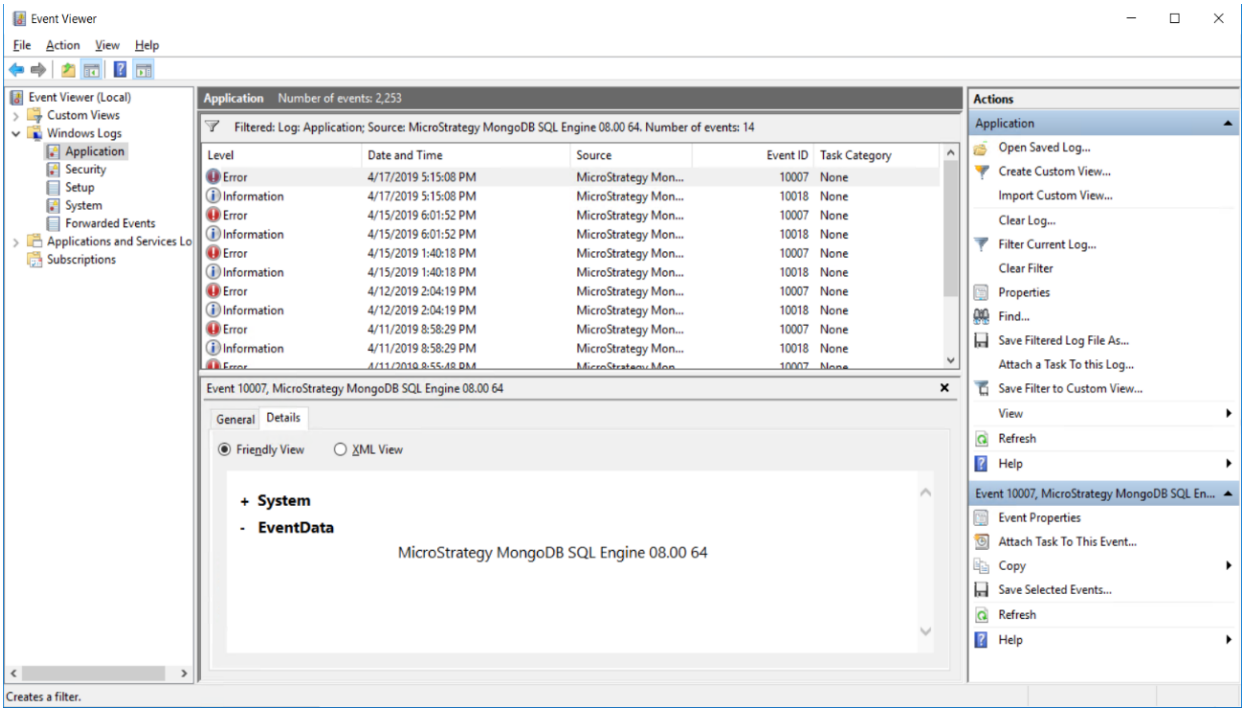
8 You can display events based on specific criteria. To do this, in the right pane, click **Filter Current Log**.



9 From the **Event Sources** drop-down list, select **MicroStrategy MongoDB SQL Engine** check box.



10 Click **OK**. All events generated by the MicroStrategy MongoDB SQL Engine application are displayed. Scan the list and review the details of any errors.



6

Monitoring and Optimizing the Infrastructure

InfiniRec's infrastructure has been sized appropriately, all required connections have been established, and operating procedures have been documented. At this point, your main task is to ensure that the environment continues to operate as expected. To keep the infrastructure running smoothly, you are ready to develop a set of standards to help administrators monitor and optimize the infrastructure components.

In this chapter, you will learn to create standards for the following monitoring and optimization tasks:

- **Hardware monitoring**: create alerts and use monitoring tools to ensure the reliability of hardware components.
- **Network monitoring**: monitor network utilization and bandwidth to ensure that the infrastructure can accommodate the required flow of data.
- **Troubleshooting**: create the appropriate log files to aid in troubleshooting tasks when something goes wrong.

Discovering and fixing hardware deficiencies

To ensure that MicroStrategy services operate optimally without interruption, the machines where the services are installed must have adequate hardware resources. If hardware components are overloaded, MicroStrategy services may not be able to operate, and users will not be able to analyze data, initiate administrative operations, or perform other functions on the MicroStrategy platform.

Creating alerts to discover hardware problems

To ensure that server hardware components function as expected, you can create server alerts and notifications. For example, you can create alerts that notify administrators when key hardware parameters exceed predefined threshold limits. Parameters can include CPU usage, memory usage, I/O requests, free disk space, file descriptor counts, and so on. You can also create alerts based on network utilization parameters like bandwidth and network utilization.

The following image contains a list of alerts that you might create using SolarWinds.

Component	Monitoring platform	Alert
MicroStrategy Intelligence Server	SolarWinds	Service Down
Server downtime	SolarWinds	Host Down
MicroStrategy Listener	SolarWinds	Service Down
PDF Export – TCP port 20100	SolarWinds	Service Down
Web Server	SolarWinds	Service Down
Collaboration server port 3000	SolarWinds	Service Down
Mongo DB	SolarWinds	Service Down
Disk space	SolarWinds	Less than 10%
CPU utilization	SolarWinds	CPU 60% - Condition must exist for more than 2 min

Based on your experience with your hardware configuration, develop standards and protocols that identify key server parameters and convey guidelines to help administrators create alerts that identify problematic resource utilization. The hardware monitoring and alerting standards that you create might include the following monitoring and alerting scenarios:

- To ensure that the Intelligence Server, Web Server, and Mobile Server machines are operating with sufficient resources, create monitoring alerts on hardware utilization statistics. For example, you can monitor and create alerts on CPU usage, memory utilization, and Input/Output (I/O) requests.
- To proactively prevent system crashes due to the lack of disk space, monitor and set alerts on the server's free disk space. For example, use a tool like Netwrix Disk Space Monitor to create alerts.
- Consistent increases in the number of file descriptors in a Linux environment may be symptomatic of an operating system or hardware problem. To monitor file descriptor counts, create a script in the Linux environment where Intelligence Server is installed.
- To ensure that the network has adequate bandwidth to transfer data between servers and clients, monitor and set alerts on network utilization. For example, you can monitor using a network monitoring tool like NetPerf.

Which monitoring tools do you have experience with?

What are some essential features for a monitoring tool?

Installing and maintaining monitoring tools

Server monitoring and alerting tools are used to continually check the values of key hardware and network utilization parameters and send notifications to an administrator when the system approaches undesirable operating conditions. For

example, the following image displays some of the third-party monitoring tools you may choose to install in your environments:

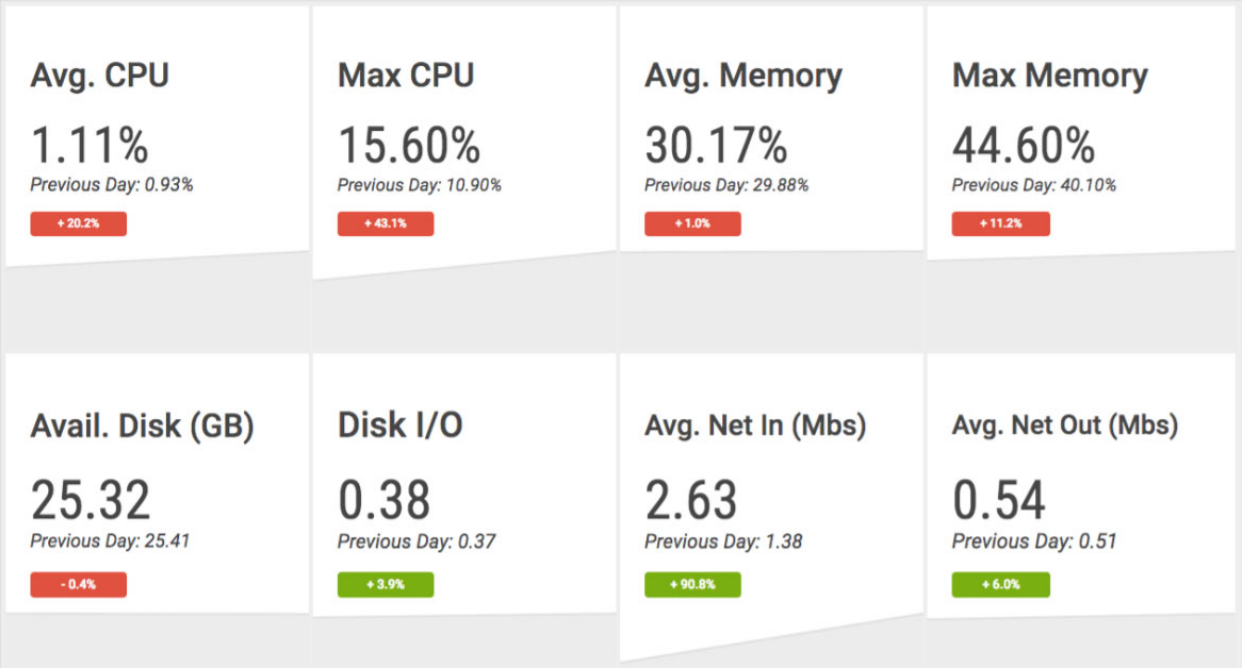
As part of your guidelines, include the following information to help administrators install and maintain server monitoring and alerting tools:

- Specify the tools that are used to monitor server hardware and operating system components such as CPU usage, memory usage, I/O requests, free disk space and file descriptor count (Linux). Include installation, upgrade, and maintenance instructions for these tools. For example, you might stipulate that Heartbeat Monitor must be installed on all server machines.
- Specify the tools to be used to estimate network utilization and bandwidth. For example, you might require administrators to use NetPerf to monitor the network.

Reviewing hardware performance to ensure reliability

Environment reliability defines the availability and robustness of each system component. Reliability is an important aspect to consider in your infrastructure because MicroStrategy users expect to have access to data at all times, regardless of external factors such as hardware malfunctions or inadequate resources. For

example, the following dossier displays key performance indicators for hardware resources.



To ensure that data analysis through the MicroStrategy platform is available to users when they need it, create standards that ensure environment reliability. For example, your standards might include the following hardware performance assessment practices:

- Establish a list of approved applications to be used by infrastructure administrators to monitor hardware performance. For example:
  - Use SolarWinds to monitor the uptime for servers and services.
  - Use New Relic to monitor memory utilization.
  - Use CloudWatch to monitor hardware utilization for your MicroStrategy on AWS environments.
  - Use Netwrix to monitor disk utilization.
- Conduct regular operational reviews of the operating system and the Network and File Systems. To perform a comprehensive review, analyze the Performance Monitor logs (Windows) and NMON Analyzer logs (Linux) to examine operating system, network, and file system statistics. Periodically verify that key hardware parameters consistently fall within the established performance benchmarks. For example, track the following hardware statistics:

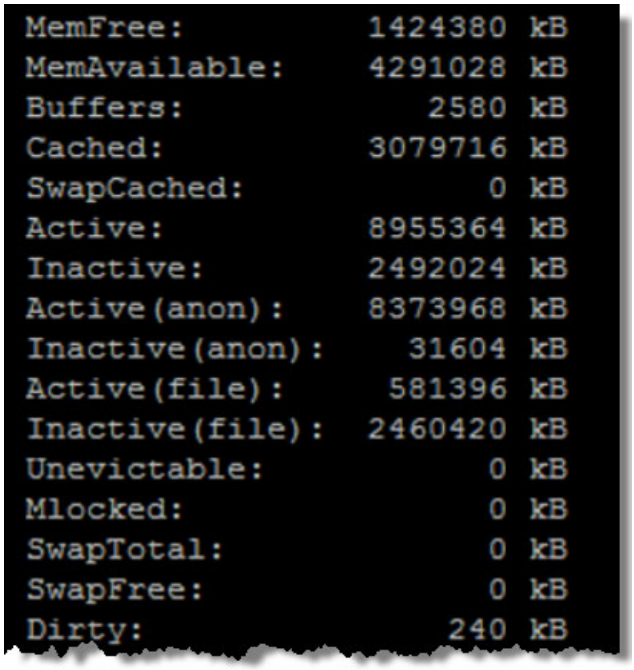
    - CPU utilization: the proportion of CPU processing power utilized by applications.
    - Memory utilization: the proportion of available memory utilized by applications.
    - I/O requests: the number of read and write requests per second.
    - Free disk space: the available amount of hard disk space in a given directory.
    - File descriptor count: the number of files opened by applications in the Linux operating system.
    - Uptime percentage: the proportion of time that a server or service is available.
- If any hardware performance statistics are measured outside the desired benchmarks, perform root cause analysis to identify a fix. For example, if memory utilization is approaching 100%, examine the applications that are consuming, determine whether any programs are consuming more memory than expected, and then identify a fix. You might choose to add memory to a machine or move an application to a different machine.

Monitoring memory utilization in Linux to avoid crashes

To accommodate high concurrency or a large number of complex reports, the Intelligence Server machine must have an adequate amount of memory to prevent swapping between physical memory (RAM) and virtual memory. To ensure that your memory hardware meets demands, you must monitor memory utilization and ensure that performance expectations are met.

To track memory utilization in Linux, you can use meminfo to display a snapshot of the physical RAM and the virtual memory. To do this, you can use the following command in your Linux environment:

cat /proc/meminfo



You can also obtain real-time memory statistics by executing sar, as in the following example:

sar <seconds to sample> <iterations> -rRB

This sar command uses the rRB options. You can use the following options in the sar command:

- -r reports the following swap utilization and memory statistics:
  - **kbmemfree**—Amount of free memory in kilobytes
  - **kbmemused**—Amount of memory used in kilobytes
  - **%memused**—Percentage of used memory
  - **kbbuffers**—Amount of memory used as buffers in kilobytes
  - **kbcached**—Amount of memory used to cache data in kilobytes
  - **kbswpfree**—Amount of free swap space in kilobytes
  - **%swpused**—Percentage of swap space used
  - **kbswpcad**—Amount of cached swap memory in kilobytes
- -R reports the following memory usage statistics:

  - **Frmpg/s**—Number of memory pages freed by the system per second. A negative value signifies the number of pages allocated by the system.
  - **Bufpg/s**—Number of additional memory pages used as buffers by the system per second. A negative value signifies that fewer pages are used as buffers by the system.
  - **Campg/s**—Number of additional memory pages cached by the system per second. A negative value signifies fewer pages in the cache.

- -B reports the following paging statistics:
  - **pgpgin/s**—Number of kilobytes paged in from disk per second
  - **pgpgout/s**—Number of kilobytes paged out to disk per second
  - **fault/s**—Number of page faults per second
  - **majflt/s**—Number of major page faults per second. This includes faults which require loading a memory page from disk.

  The faults (fault/s) and major page faults (majflt/s) metrics should consistently be as close to 0 (zero) as possible. High values for these metrics indicate a memory performance issue.

The following example provides sample data for the sar utility with the –r and –B options:

sar 5 2 -rB

```
[mstr@env-74522laiouse1 ~]$ sar 5 2 -rB
Linux 3.10.0-514.16.1.el7.x86_64 (env-74522laiouse1)    10/17/2017      _x86_64_        (2 CPU)

11:14:37 PM  pgpgin/s pgpgout/s   fault/s  majflt/s  pgfree/s pgscank/s pgscand/s pgsteal/s    %vmeff
11:14:42 PM      0.00     21.00     15.20      0.00    151.80      0.00      0.00      0.00      0.00

11:14:37 PM kbmemfree kbmemused  %memused kbbuffers  kbcached  kbcommit   %commit  kbactive   kbinact   kbdirty
11:14:42 PM   1539876  11855796     88.50      2580   2837796  24617304    183.77   9200760   2281320       212

11:14:42 PM  pgpgin/s pgpgout/s   fault/s  majflt/s  pgfree/s pgscank/s pgscand/s pgsteal/s    %vmeff
11:14:47 PM      0.00      9.60     21.40      0.00     19.40      0.00      0.00      0.00      0.00

11:14:42 PM kbmemfree kbmemused  %memused kbbuffers  kbcached  kbcommit   %commit  kbactive   kbinact   kbdirty
11:14:47 PM   1539892  11855780     88.50      2580   2837800  24617304    183.77   9200748   2281324       220

Average:     pgpgin/s pgpgout/s   fault/s  majflt/s  pgfree/s pgscank/s pgscand/s pgsteal/s    %vmeff
Average:         0.00     15.30     18.30      0.00     85.60      0.00      0.00      0.00      0.00

Average:    kbmemfree kbmemused  %memused kbbuffers  kbcached  kbcommit   %commit  kbactive   kbinact   kbdirty
Average:      1539884  11855788     88.50      2580   2837798  24617304    183.77   9200754   2281322       216
[mstr@env-74522laiouse1 ~]$
```

Ideally, swapping should not occur on the Intelligence Server machine. The kilobytes of swap used (kbswpused) and percentage of swap used (%swpused) statistics indicate whether the swap space was sized correctly, and whether an adjustment is required. If the system is continuously swapping, your Intelligence Server machine may require a memory upgrade to increase performance.

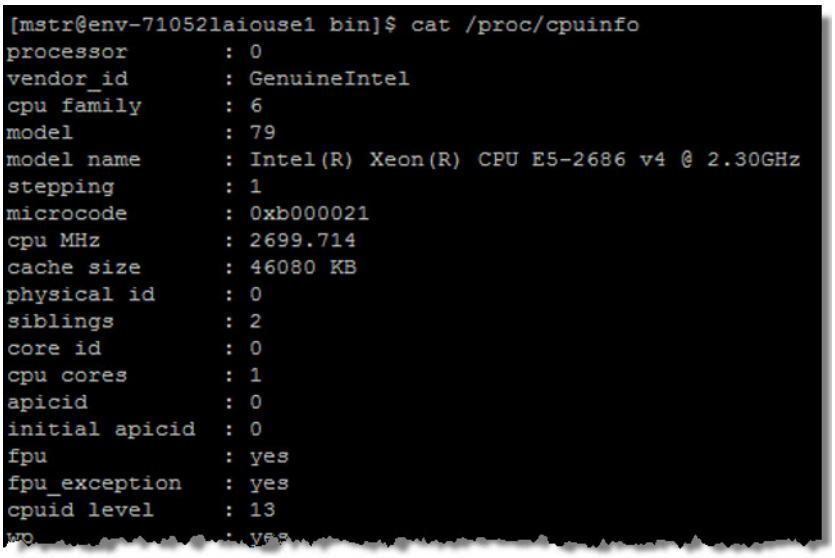
Monitoring and tuning processor performance to meet user demand

The Intelligence Server machine processor (CPU) is a main factor in processing analytics requests. The processor power dictates its ability to handle concurrency and complex reports in your environment. It is essential to monitor processor utilization to ensure that it meets the demands placed on it by MicroStrategy users.

In a Linux environment, you can use the following commands to monitor and tune the processor on your Intelligence Server machine:

- To display information about the processor, use the following command:

  cat /proc/cpuinfo



- To monitor CPU utilization on the Intelligence Server machine, execute the following command:

  sar -u

  The report outputs the following metrics:

  - **%user**—Percentage of CPU utilization executing at the user level (application)
  - **%nice**—Percentage of CPU utilization executing at the user level with nice priority
  - **%system**—Percentage of CPU utilization executing at the system level (kernel)

- **%iowait**—Percentage of time that the CPU was idle while there were outstanding I/O requests
- **%idle**—Percentage of time that the CPU was idle while there were NO outstanding I/O requests

You can also report on per-processor statistics by executing the following command:

sar –P CPU

or

sar - P ALL

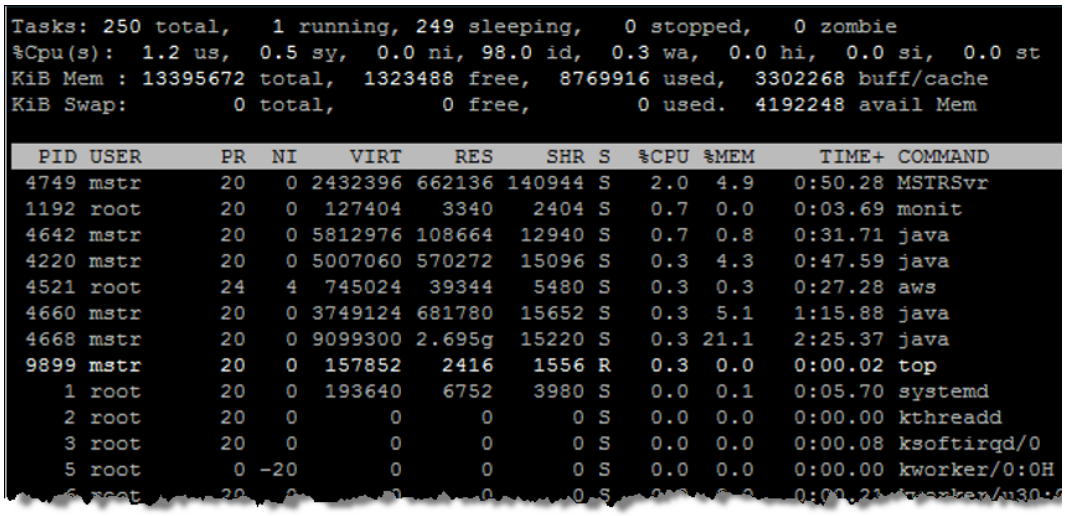
You can specify the desired CPU (for example, sar -P 0), or use the ALL keyword to report statistics for each individual processor, as well as globally for all processors. When viewing the output, note that 0 (zero) signifies the first processor.

System time, the measure of CPU utilization for kernel processes, should never exceed 10%. If this value exceeds the 10% mark, there may be an issue with the Linux operating system itself, and further investigation is required.

Total CPU utilization, that is, the sum of user, system, and I/O wait utilization, must not exceed 80-85%.

- If the processor is unable to keep up with demand, you can identify and terminate processes that are using the most processor resources. To do this, execute the following command:

top



Ensure that the %CPU usage for the MSTRSvr and other processes is 80% or less.

If a CPU bottleneck has been detected through the use of Linux's activity reporting utilities, you should also look into updating drivers, upgrading hardware, and binding processes to specific processors. Depending on the situation encountered, you may use one or a combination of the above options to resolve your bottleneck.

After you execute top, you can press **CTRL+ Z** to return to the $ prompt.

Monitoring disk space utilization to avoid slowdowns

Hard disk space on the Intelligence Server machine is required to store critical MicroStrategy platform operation files. To store the required files, ensure that the available disk space on the Intelligence Server machine is at least three times the amount of RAM available to Intelligence Server.

To identify the available disk space on a Linux machine, execute the df or df –k command. This command displays the amount of disk space available on the file system, containing each file name argument in a block size of 1K. It reports the amount of disk space used by the specified files, and by each directory in the hierarchies rooted at the specified files.

If no file name is given, the space available on all currently mounted file systems is shown. Disk space is shown in 1K blocks by default.

For example, the following image shows the output when the following commands are used:

df

or

df -k

```
[mstr@env-71052laiouse1 bin]$ df
Filesystem     1K-blocks     Used Available Use% Mounted on
/dev/xvda2      20959212 10054228  10904984  48% /
devtmpfs         7709536        0   7709536   0% /dev
tmpfs            6697836        4   6697832   1% /dev/shm
tmpfs            6697836    17404   6680432   1% /run
tmpfs            6697836        0   6697836   0% /sys/fs/cgroup
/dev/xvdb2      24402288  1105528  23296760   5% /opt/usher
/dev/xvdc       52403200  1860244  50542956   4% /opt/mysql
/dev/xvdb3     265613604 11209956 254403648   5% /opt/mstr
/dev/xvdb1      24402288  2779468  21622820  12% /opt/apache/tomcat
tmpfs            1339568       12   1339556   1% /run/user/1002
[mstr@env-71052laiouse1 bin]$
```

You can use the following options with the df command:

- **-a, --all**

  Includes file systems having 0 blocks

  Example:

  df -a

  or

  df --all

- **-h, --human-readable**

  Displays sizes in human readable format such as, 1K, 234M, or 2G

- **-H, --si**

  Displays sizes in human readable format but uses powers of 1000 (not 1024)

- **-i, --inodes**

  Lists inode information instead of block usage

- **-l, --local**

  Limits listing to local file systems

- **-P, --portability**

  Uses the POSIX output format

- **-T, --type=TYPE**

Limit listing to file systems of type TYPE

Example:

df --type=xfs

- **-T, --print-type**

  Displays file system type

- **-x, --exclude-type=TYPE**

  Limits listing to file systems not of type TYPE

Use these commands to determine the amount of hard disk space available on your machine, and to identify the directories where you can delete files to make room.

Increasing hardware resources to accommodate resource-intensive operations

Environment resource requirements refer to each system component's capacity to handle the demand placed on it. The required processor and memory resources on a given infrastructure machine depend on factors such as the MicroStrategy platform components installed on the machine, operating system, third party software, number of concurrent users, and types of operations performed on the machine.

For example, a server that runs the Intelligence Server should be able to accommodate the number of users who run reports and perform other platform functions during peak utilization hours. To ensure that the server performs optimally and appropriately handles the demand placed on it by user activity and MicroStrategy platform operations, create standards that help infrastructure administrators conduct regular reviews of infrastructure machines to identify resource deficiencies.

Your standards might include the following tasks related to hardware optimization:

- Work with the Platform Administrator to determine if the processor speed or number of processing units needs to be increased.

  For example, if the processor is consistently running at high capacity (greater than 80%), consider procuring a faster processor or increasing the number of processing units.

- Work with the Platform Administrator to determine if the MicroStrategy Intelligence Server machine memory needs to be increased.

  For example, if the memory counters indicate that Intelligence Server is consistently using more than 80% of the Operating System memory resources, or the Intelligence Server has a higher number of memory contract request rejections, consider increasing memory on the Intelligence Server machine.

Exercise 6.1: Monitoring hardware utilization in Linux

Monitoring hardware utilization enables you to understand how your infrastructure is being used, and determine whether optimization activity or additional hardware allocation is required to deliver the performance that MicroStrategy users demand.

In this exercise, you will monitor memory, processor, and disk utilization on your Linux server.

Monitor hardware utilization

1 Open the RDP connection to your Windows cloud machine.

2 From the desktop, double-click the **hosts.txt** file and open it with **Notepad**.

3 At the bottom of the file, copy the Intelligence Server machine IP address.

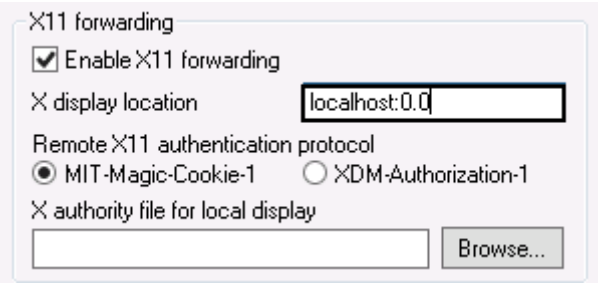
4 From the desktop, double-click **PuTTy**.

5 In the Category pane, click **Session**.

6 In the **Host Name** box, paste the Intelligence Server machine IP address.

7 In the Category pane, under Connection, expand **SSH** and click **X11**.

8 Select the **Enable X11 forwarding** check box.



9 In the **X Display location** box, type **localhost:0.0**

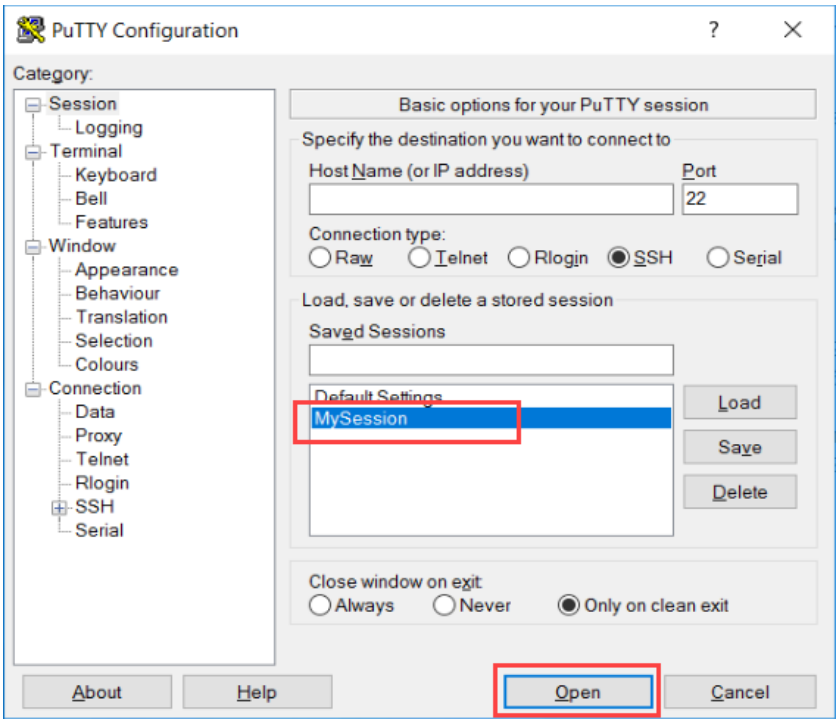
10 In the Category pane, click **Session**.

11 In the **Saved Session** box, type **MySession**.

12 Click **Save**.

13 In the Saved Sessions area, click **MySession**.

14 Click **Open**.



15 In the Security Alert window, click **Yes**.

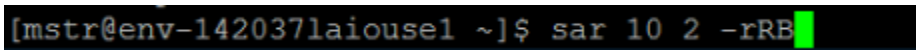
16 In the PuTTy window, log in with the credentials in your MicroStrategy Cloud email.

Monitor memory utilization

17 At the $ prompt, type **cat /proc/meminfo** and press **Enter**.

18 How much free memory does this machine have?

19 At the $ prompt, type **sar 10 2 -rRB** and press **Enter**. Two 10-second cycles are sampled.

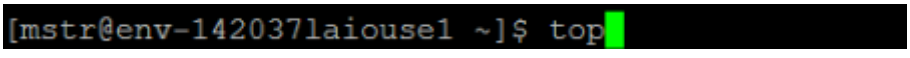


20 What percentage of memory was used during each 10-second period? What action would you take if memory utilization was approaching 100%?

21 Were there any major page faults during either period? What could you do to reduce page faults?

Monitor processor utilization

22 At the $ prompt, type **top** and press **Enter**.

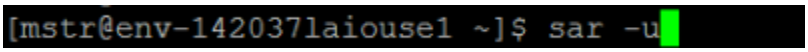


23 What percentage of processing power is consumed by the MSTRSvr process?

24 What process is utilizing the highest percentage of processing power?

25 Press **Ctrl** + **Z**.

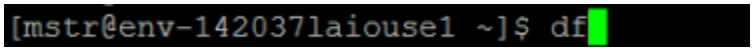
26 At the $ prompt, type **sar -u** and press **Enter**.



27 What percentage of processing power is being utilized at the user level? How can you tell if the CPUs are not keeping up with demand? What would you do to alleviate high CPU utilization?

Monitor disk utilization

28 At the $ prompt, type **df** and press **Enter**.



29 How many file systems are there on the machine?

30 Which file system has the highest rate of utilization? What action would you take if one of the file systems was approaching 100% utilization?

Fine-tuning network performance to accommodate data flow

Environment performance refers to the system's ability to respond to demands, based on a set of benchmarks and guidelines. Ensuring that your system components are able to meet MicroStrategy platform, third-party software, and end user demands is critical to analytics operations and user buy-in. A system that cannot accommodate its user base causes frustration, and will eventually be abandoned.

As the System Administrator, your goal is to create standards that help infrastructure administrators understand performance considerations and tune the system to keep up with demand. Your performance optimization guidelines may include the following:

- Coordinate with the Platform Administrator to tune Intelligence Server configuration settings, including minimizing the file descriptor usage on the Linux operating system for the Intelligence Server process. To do this:
  - Optimize the number of database connections

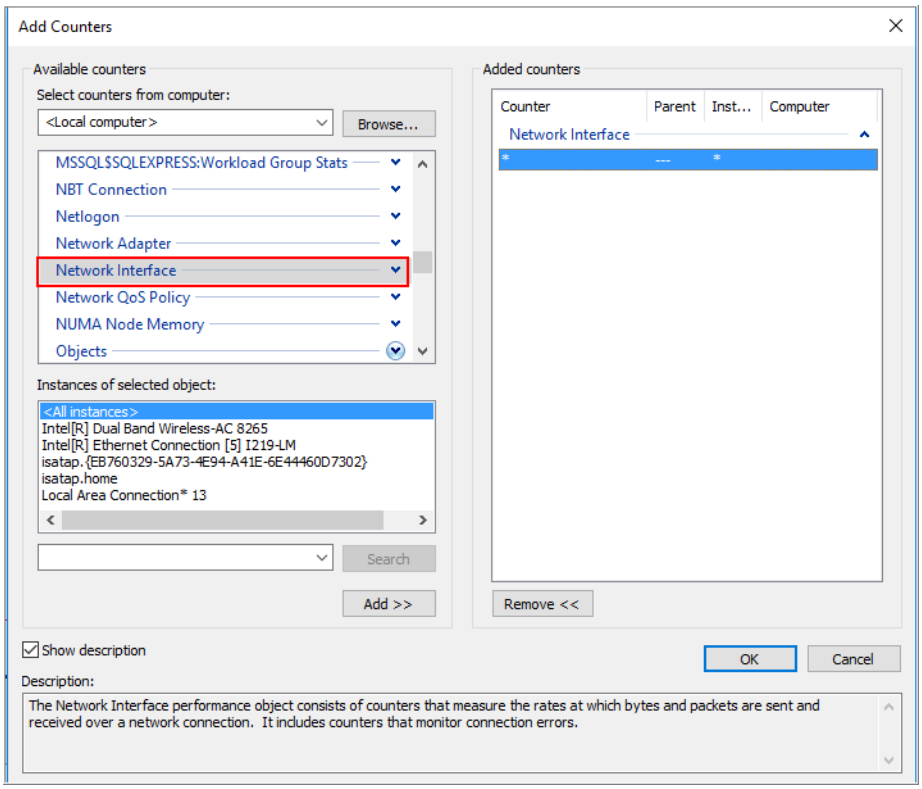
    - Optimize the number of network connections
    - Use an effective Cache, Cube and History List file management strategy to balance the number and frequency of Cache, Cube and History List file hits
    - Minimize the number of log (diagnostic) files
- Optimize network utilization. For example, if the network throughput as a percent of network bandwidth is consistently greater than 60%, do the following:
    - Increase the network bandwidth
    - Work with Platform Administrator to tune MicroStrategy server settings and alleviate high network usage. To do this:
        - Increase the amount of element and dataset caching
        - Employ incremental fetch for result sets
    - Decrease the proximity between the Intelligence Server and the Web Server machines to optimize network usage
    - Decrease the proximity between the Intelligence Server and the databases (metadata and warehouse) to optimize network usage
    - Adjust the MicroStrategy Query Engine settings to increase data chunking across the network and fetch additional rows through the ODBC layer

Investigating network bandwidth

To determine whether your system performance is negatively impacted by your current network configuration, investigate your network's capacity utilization. To do this, use network monitoring tools such as NetPerf or Windows Performance Monitor.

For example, in a Windows environment, you can use the Windows Performance Monitor to gauge traffic in the Network Interface object. You can monitor the Total bytes/sec counter, which measures traffic as a percent of your network's bandwidth. You can track this measurement at various times with varying

concurrency rates in a test environment to determine whether network bandwidth can accommodate demand.



To calculate the network capacity utilization percentage, divide the total capacity, in bits per second, by (Total bytes per second * 8). (Multiply the Total Bytes per second by 8 because 1 byte = 8 bits.) You can also use the Current Bandwidth counter in Performance Monitor, which provides an approximation of the network's total capacity.

As part of your performance monitoring standards, identify the network monitoring procedures that infrastructure administrators should follow. For example, you might include the following tasks:

- Specify the preferred applications for network monitoring. For example:
  - Use Wireshark for TCP/IP tracing to troubleshoot network problems.
  - Use New Relic to monitor network latency between servers and clients.
- Specify the network counters that should be tracked. For example, network bandwidth utilization should not approach 100%.

Exercise 6.2: Testing network bandwidth through Performance Monitor

Your network bandwidth dictates the speed at which data is delivered to MicroStrategy end users. To ensure that network bandwidth can adequately meet user demand, you must measure it and determine whether utilization is reaching capacity.

In this exercise, you will use the Windows Performance Monitor to inspect network bandwidth and utilization.

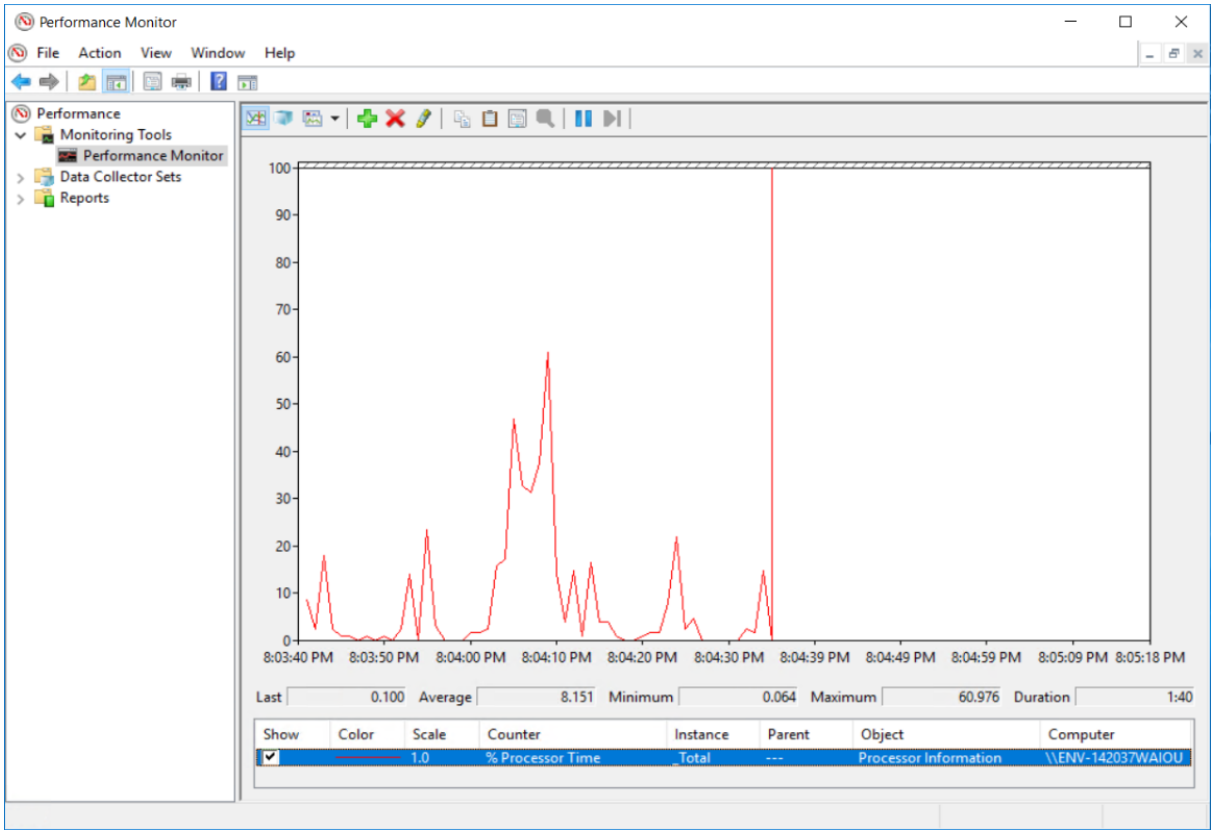
Inspect network bandwidth and utilization

1 Access the RDP connection to the Windows machine in the cloud.

2 From the task bar, in the **Search** box, type **Performance Monitor**.

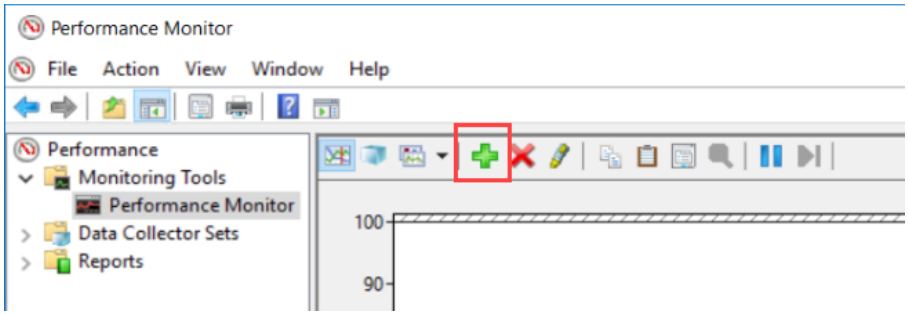
3 From the search results, click **Performance Monitor**.

4 In the left pane, expand **Monitoring Tools** and click **Performance Monitor**.



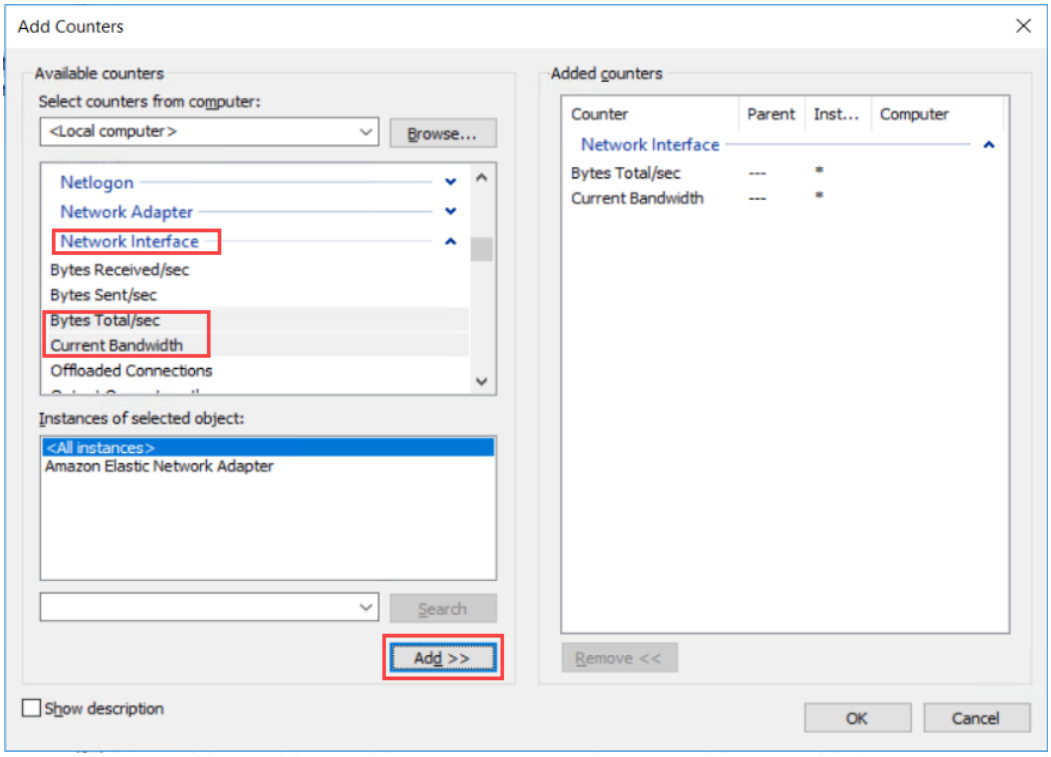
Add network counters to the graph

5 At the top of the right pane, click **Add**. The Add Counters window opens.



6 In the left pane, click the down arrow next to **Network Interface**.

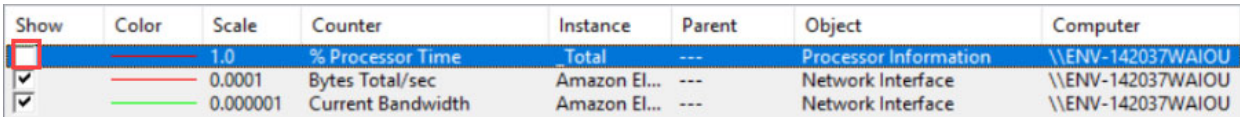
7 Hold **Ctrl** and select **Bytes Total/sec** and **Current Bandwidth**.



8 Click **Add**. The Network Interface counter is added to the Added Counters pane.

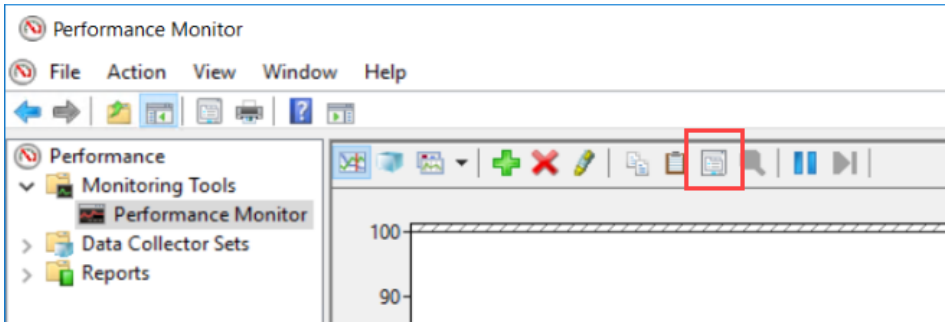
9 Click **OK**.

10 In the bottom pane, clear the **% Processor Time** check box.

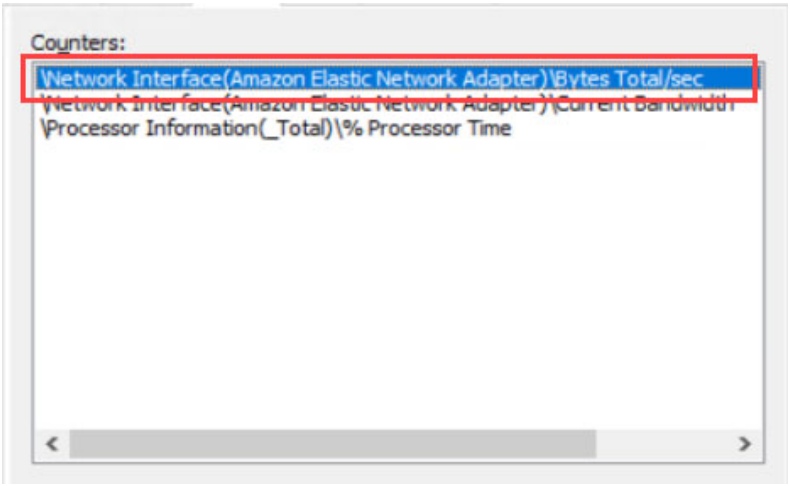


Modify the graph display

11 At the top of the screen, click **Properties**. The Performance Monitor Properties window opens.



12 In the Data tab, in the Counters area, click **Bytes Total/sec**.



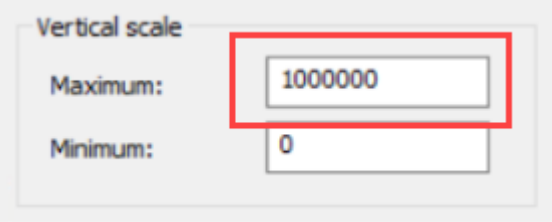
13 From the **Scale** drop-down list, select **1.0**.

14 From the Counters area, click **Current Bandwidth**.

15 From the Scale drop-down list, select **1.0**.

16 Click the **Graph** tab.

17 In the Vertical Scale area, in the **Maximum** box, type **1000000** (one million).



18 Click **OK**. The graph displays your current bandwidth and the total amount of data on the network interface.

Notice that the bandwidth is at the extreme top of the graph, while the total of bytes is at the extreme bottom. There is minimal data moving through this part of the network.

19 Open a browser and perform a network-intensive activity like video streaming.

How does the graph change? Is the total data approaching the bandwidth capacity? If the total data and bandwidth capacity were in close proximity to each other, how would you modify your network to accommodate more data?

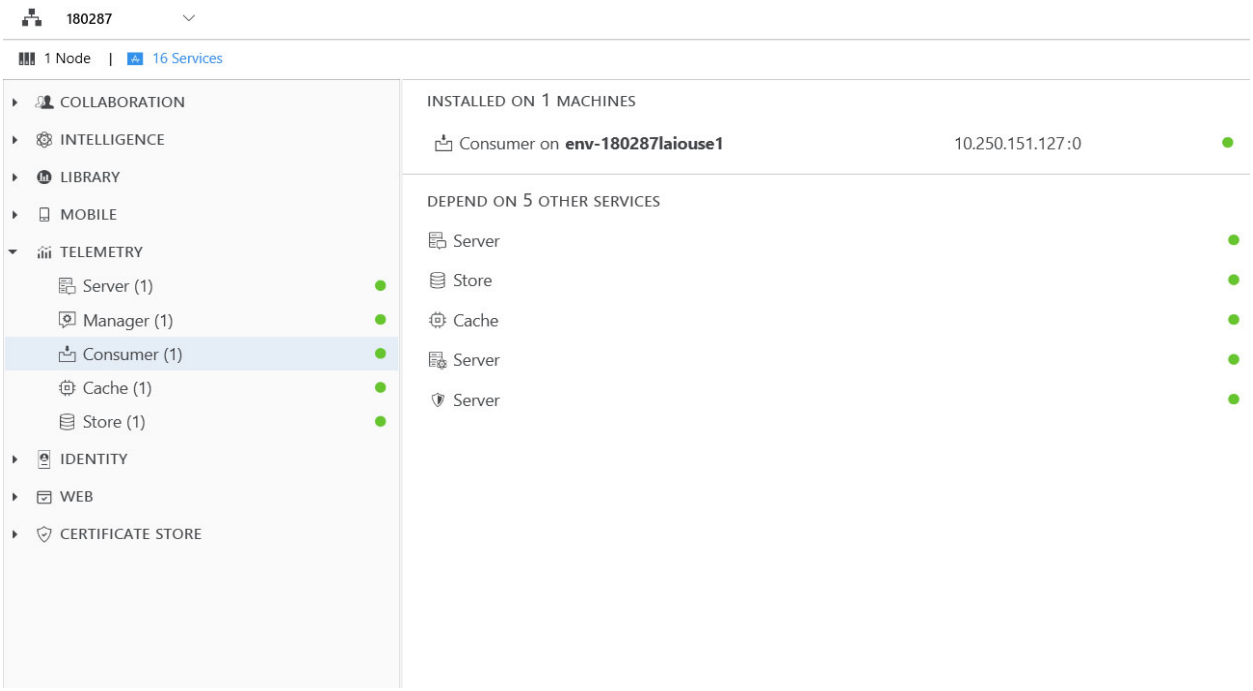
Providing troubleshooting support to the Platform Administrator

Despite careful planning and diligent monitoring, your infrastructure components are likely to encounter operational problems or even failures. When these issues arise, your organization's infrastructure administrators must be prepared to support the Platform Administrator in troubleshooting infrastructure components. To prepare administrators for troubleshooting responsibilities, develop troubleshooting protocols.

Monitoring MicroStrategy servers and services

Your MicroStrategy environment likely consists of several servers that house distinct MicroStrategy components. Instead of logging into each machine individually, you can remotely monitor the status of MicroStrategy servers and services from a single location using MicroStrategy Workstation. You can view the

status of each service, determine whether it is running, view its dependencies, and start or stop the service, as displayed in the following image.



Gauge system health by viewing the color icons displayed next to each node or service and determine whether they need to be restarted. Each node and service is displayed with one of the following color status icons:

- **Green**: The node or service, and underlying services, are currently running.
- **Yellow**: The node or service contains at least one underlying service that is currently not running.
- **Red**: The node or service is currently not running.

Exercise 6.3 Monitoring services in Workstation

InfiniRec has installed MicroStrategy components on multiple servers. You want to monitor the installed services to ensure they are running as expected and restart any services that might be down. You could log into multiple servers individually to monitor service status, but MicroStrategy Workstation enables you to monitor multiple nodes and their services from a single interface.

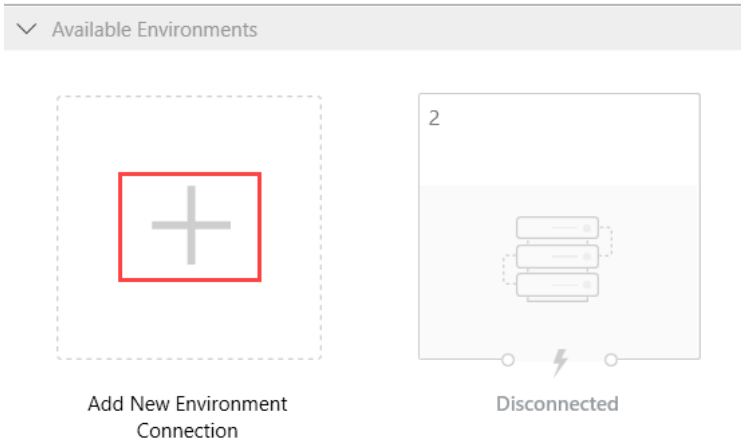
In this exercise, you will connect Workstation to an environment to view the environment topology and monitor nodes and services.

Connect Workstation to your Environment

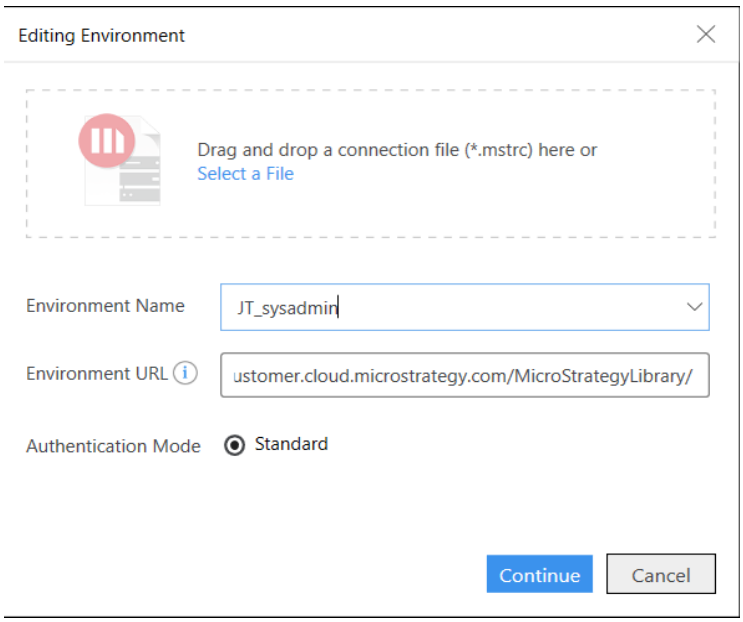
1 From the Windows desktop, double-click **MicroStrategy Workstation**.

2 From the left pane, click **Environments**.

3 In the right pane, click **Add New Environment Connection**.



4 In the Add New Environment Connection window, type the following information:



a In the **Environment Name** box, type ***your initials*_sysadmin**. For example, JT_syadmin.

b In the **Environment URL** box, type your environment's Library Web Server URL in the following format:

https://env-XXXXXX.customer.cloud.microstrategy.com/MicroStrategyLibrary/

where **XXXXXX** is the unique environment ID listed in your MicroStrategy Cloud email.

c In the **Authentication Mode** area, click **Standard**.

5 Click **Continue**.

6 In the Connect to Environment window, type the **User Name** and **Password** from your MicroStrategy Cloud email.

7 Click **Connect**.

8 In the Select Applications window, click the **MicroStrategy Tutorial** check box and click **OK**.

Now that you have connected Workstation to your environment, you are ready to view the environment topology and the status for each node and service.

Monitor node and service status

1. In the left pane, under Manage, click **Topology**. Your environment is displayed in the right pane. You have one node with 16 services installed on it.
2. Click the down arrow at the bottom right corner of the node. The installed services and their statuses are displayed.
3. Hover your cursor over the red indicators to view possible reasons for the stopped status.
4. Right-click **Identity Telemetry** and click **Start**.l
5. In the Log Into Machine via SSH window, type the **Username** and **Password** from your MicroStrategy Cloud email.
6. Click **Log In**.
7. In the Start Identity Gateway window, click **OK**.
8. In the START Command message window, click **OK**.
9. At the top of the screen, click **Refresh**.

   Identity Telemetry now has a status of Running.
10. Right-click **Identity Telemetry** and click **Stop**.
11. In the STOP Command message window, click **OK**.
12. At the top of the screen, click **Refresh**. The Identity Telemetry service is now stopped.

Troubleshooting Intelligence Server performance problems and crashes

The standards and protocols you create should help infrastructure administrators support the Platform Administrator in resolving issues impacting the stability, performance, or security of the environment. Your organization's protocols might include the following tasks to help administrators restore normal operations and investigate root causes in the event of a server problem:

Best Practice

- In the event of an Intelligence Server crash or abnormal shutdown, perform the following tasks:
  - Ensure that the operating system has been configured to generate core files.
  - Work with the Platform Administrator to immediately restart the Intelligence Server.
  - Upload core files to a location where they can be accessed by the Platform Administrator, who will send the files to MicroStrategy Technical Support for root cause analysis.

Best Practice

- If the Intelligence Server hangs or becomes unresponsive:
  - Ensure that the collection of pstacks (Linux) or hang mode dumps (Windows) are enabled.
  - Work with the Platform Administrator to immediately restart the Intelligence Server.
  - Upload the pstacks or dumps to a location where it can be accessed by the Platform Administrator, who sends the files to MicroStrategy Technical Support for root cause analysis.

Best Practice

- If the Intelligence Server shuts down due to memory depletion:
  - To prevent future Intelligence Server memory failures, work with the Platform Administrator to implement MicroStrategy's Memory Contract Management governors.
  - Collect operating system performance logs. For example, you can use Performance Monitor logs in a Windows environment or the NMON Analyzer logs in a Linux/UNIX environment.
  - Upload the performance logs to a location where it can be accessed by the Platform Administrator, who sends the logs to MicroStrategy Technical Support for root cause analysis.

Configuring Linux to generate core dump files

When the Intelligence Server crashes, a core dump file is generated by the operating system for the Intelligence Server (MSTRSvr) process. A core dump is a snapshot of the Intelligence Server in memory at the time of the crash.

By reading the core dump file, MicroStrategy Technical Support can investigate the actions taken prior to the crash, and obtain information about the report or

user that caused the crash. A call stack can also be retrieved from the core dump file to understand the functions used at the time the SSD was written.

By default, the core dump file is created under the IntelligenceServer subdirectory under the home directory. To accommodate the large file size and to avoid truncation, ensure there is sufficient hard disk space in that location. Truncated core dump files cannot be used for troubleshooting purposes.

To ensure that a core dump file needed for further investigation is always automatically generated, configure the following system settings:

The following settings are based on standard out-of-the-box systems. You may need to modify these settings to account for your unique environment's configuration.

1 Set the limits for the Intelligence Server user account so that core files can be created successfully. When the size of the core dump file is unpredictable, it is recommended the limit be set to unlimited.

   You can set limits using the following ulimit command:

   ulimit -c unlimited

   or

   ulimit -f unlimited

   The -c option sets the maximum size of core files created. The -f option sets the maximum size of files created by the shell.

   The ulimit (**U**ser**Limits**) command is a built-in shell function which displays and sets user level restrictions to system resources in UNIX/ Linux environments. If incorrectly set, these limits can cause unexpected or undesired behaviors in Intelligence Server. Limits that are too low can also negatively affect performance. These UserLimits prevent single users from bottlenecking system resources.

   Run the preceding command using the root account. To verify that the settings have taken effect, execute the following command to report all current user level limits:

   ulimit -a

The following is an example of the sample output generated using the ksh shell (your output may vary when using other shell interfaces):

```
$ ulimit -a
time(cpu-seconds) unlimited

file(blocks) unlimited

coredump(blocks) unlimited

data(kbytes) unlimited

stack(kbytes) 10240

lockedmem(kbytes) 32

memory(kbytes) unlimited

nofiles(descriptors) 1024

processes 8191
```

If changes were made successfully and you are running the Intelligence Server as a service, restart the entire machine for these changes to take effect.

2 Add the following line to /etc/security/limits.conf (if it is not already there):

*** soft core unlimited**

3 Add the following parameters to /etc/sysctl.conf (if not already there):

kernel.core_pattern = core.%e.%p

fs.suid_dumpable = 2

The kernel.core_pattern parameter specifies the location and naming scheme for core files. The preceding setting will result in core files being written to the working directory of the crashing process with a name in the form of core.<process name>.<process id>. For example, in the case of an Intelligence Server (MSTRSvr process) core with process ID 12345, the filename would be core.MSTRSvr.12345 and the core file would be written to the <MSTR_HOME>/IntelligenceServer folder.

fs.suid_dumpable = 2 configures the system to log the information necessary to capture stack traces for all existing threads in a process. In addition, it logs the data sections from all loaded modules including global variables.

4 Add the following line to /etc/sysconfig/init (if not already there):

DAEMON_COREFILE_LIMIT='unlimited'

5 Run the following command to apply any changes made:

sysctl -p

6 Ensure that the home folder for Intelligence Server has enough free space to store the core dump. The expected file size is approximately the amount of memory used by the Intelligence Server process at the time of the exception. If the Intelligence Server folder is loaded using NFS, ensure that large files can be created on the file system.

Depending on your Linux version, you may be able to check if the core can be created correctly by forcing a core file to be created using the following command from the /home/IntelligenceServer subdirectory:

gencore <process id for the Server> <core file to be created>

When you execute this command, note the location of the core file created.

7 The default user profile may be set to disable the creation of core files, especially for processes launched in the initial phase (only when the Intelligence Server is registered a a service). To address this, modify the following as a root user:

/etc/init.d/mstr-InstallName-iserver-CastorServer init script

Add the following line to allow the Intelligence Server to start as a service with the core files enabled:

ulimit -c unlimited

Add the preceding line, under the line that reads:

start ()

{

InstallName is of the form user@timestamp and can be found in the MSIReg.reg file.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\MicroStrategy\DSS Suite]
"ActivationTechSupportURL"="https://licensing.microstrategy.com"
"Behavior"="/opt/mstr/MicroStrategy/mstr.bhv"
"Common Path"="/opt/mstr/MicroStrategy/install"
"FirstInstallVersion"="10.9.0046.0035"
"HFBVersion"=""
"HFRVersion"=""
"HistoryPath"="/opt/mstr/MicroStrategy/log"
"HomePath"="/opt/mstr/MicroStrategy"
"InstallName"="root@2017-10-16T17:30:55"
"InstallPath"="/opt/mstr/MicroStrategy/install"
"LicenseKey"=""
"LogPath"="/opt/mstr/MicroStrategy/log"
"RVersion"="10.9.0"
"RegPortNumber"="80"
```

Exercise 6.4: Creating a stack trace to troubleshoot an unresponsive Intelligence Server

InfiniRec's MicroStrategy users are experiencing slow performance in your production environment and the Platform Administrator wants to send a stack trace of the MSTRSvr process to the MicroStrategy Technical Support team for further investigation.

In this exercise, find the MSTRSvr process ID and then generate a stack trace. Output the stack trace to a text file that can be uploaded to a central repository where the Platform Administrator can download it.

Create a stack trace

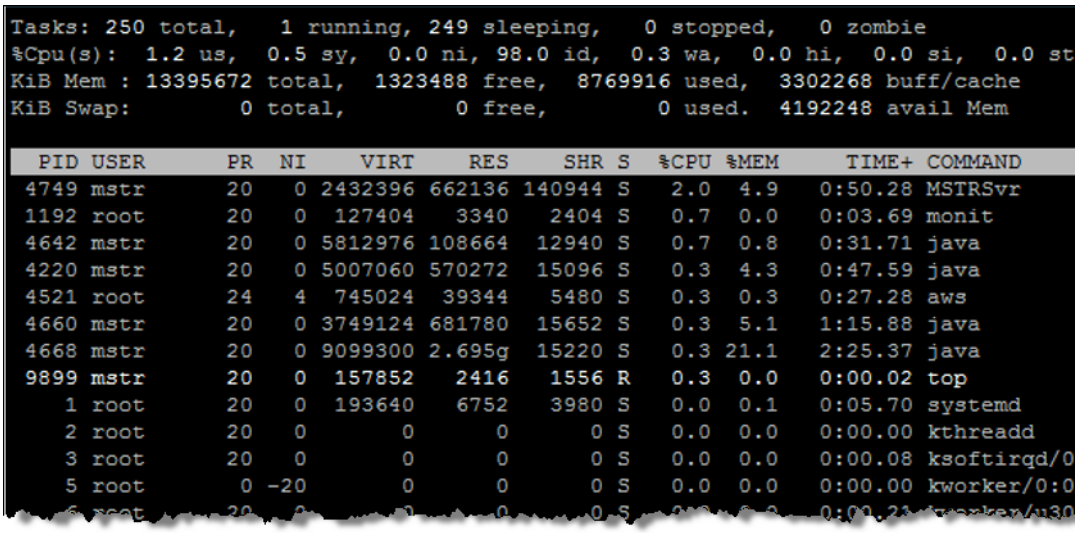
1. From the Windows desktop, double-click **PuTTy**.
2. Open MySession and log in using the credentials from the MicroStrategy Cloud email.
3. To change the current directory to the home directory, on the console, type:

   cd /home/mstr

Identify the PID

4. To identify the PID of MSTRSVR process, execute the top utility. To do this, from the $ prompt, type **top** and press **Enter**.

The top utility displays processes that are currently consuming the most CPU and memory resources.



Locate the PID for the MSTRSvr process. For example, in the preceding image, the PID is 4749.

Generate the stack trace

5 Press **Ctrl + Z**.

6 At the $ prompt, type the following command:

pstack #### > MyPstack.txt

where #### is the PID for the MSTRSvr process. For example, 4749 in the image above.

7 Press **Enter**. The stack trace is generated and output to a text file in the home directory after a few minutes.

Locate the log file

8 From the Windows desktop, double-click **hosts.txt**.

9 Select **Notepad** and click **OK**.

10 At the bottom of the file, copy the Intelligence Server IP address.

11 Minimize the hosts file.

12 From the Windows desktop, double-click **WinSCP**.

13 In the **Host Name** box, type the IP address you copied.

14 In the **User Name** box, type **mstr**.

15 In the **Password** box, type the password from your MicroStrategy Cloud email.

16 Click **Save**.

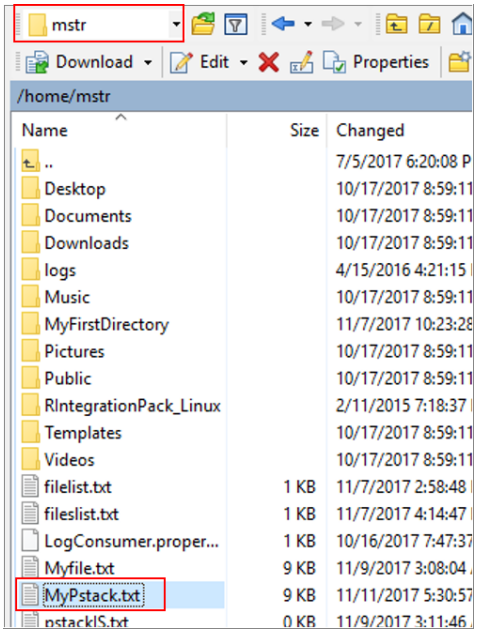
17 In the Save Session as site window, click **OK**.

18 Click **Log In**.

19 In the **Password** box, type the password from your MicroStrategy Cloud email.

20 In the right pane, navigate to **/home/mstr/**. If the MyPstack.txt file is not displayed, click **Refresh**.

21 To download the file, right-click **MyPstack.txt** and click **Download**. The file is downloaded to the Windows machine in the directory in the left pane.



Troubleshooting infrastructure problems

As MicroStrategy users interact with the MicroStrategy platform and the distributed analytics applications, they may encounter infrastructure problems that reduce performance, terminate connectivity, or produce crashes. For example, these problems can be rooted in the operation of the operating system, network, file system, hardware, directory services, or drivers.

When infrastructure issues arise, administrators are responsible for troubleshooting and fixing the problem. To ensure that administrators follow a consistent and efficient protocol when troubleshooting infrastructure problems, create standards to guide them. Your troubleshooting guidelines might include the following:

- Standard protocols to troubleshoot the operating system, network, file system, hardware, directory services, and drivers. Your protocols should include:
  - Common troubleshooting scenarios based on identified symptoms
  - The expected issue resolution timeline, according to your SLAs
  - Contact information for the appropriate provider or vendor
- Steps to troubleshoot query execution, data fetching, and connections to database servers. For example, you should include steps to retrieve and examine ODBC trace logs.

Exercise 6.5: Creating an ODBC trace log

InfiniRec's MicroStrategy users are experiencing problems connecting to a database through the MicroStrategy platform.

To help you troubleshoot the reported connection issues, you can enable ODBC trace logs, which record all ODBC calls in a text file. To mitigate the performance impact, perform a trace for a limited period of time, execute the problematic operations, and then turn the trace off. You can then scan the trace log to troubleshoot the connection problem.

In this exercise, you will turn on ODBC tracing through the ODBC.ini file, which contains the following section of tracing options:

[ODBC]

Trace=0

TraceFile=<location>/odbctrace.out

TraceDll=<MSTR_INSTALL>/install/lib/MYtrcXX.so

InstallDir=<MSTR_INSTALL>

IANAAppCodePage=106

UseCursorLib=0

Create an ODBC trace log

1 From the Windows desktop, double-click **WinSCP**.

2 Double-click the session that you previously created.

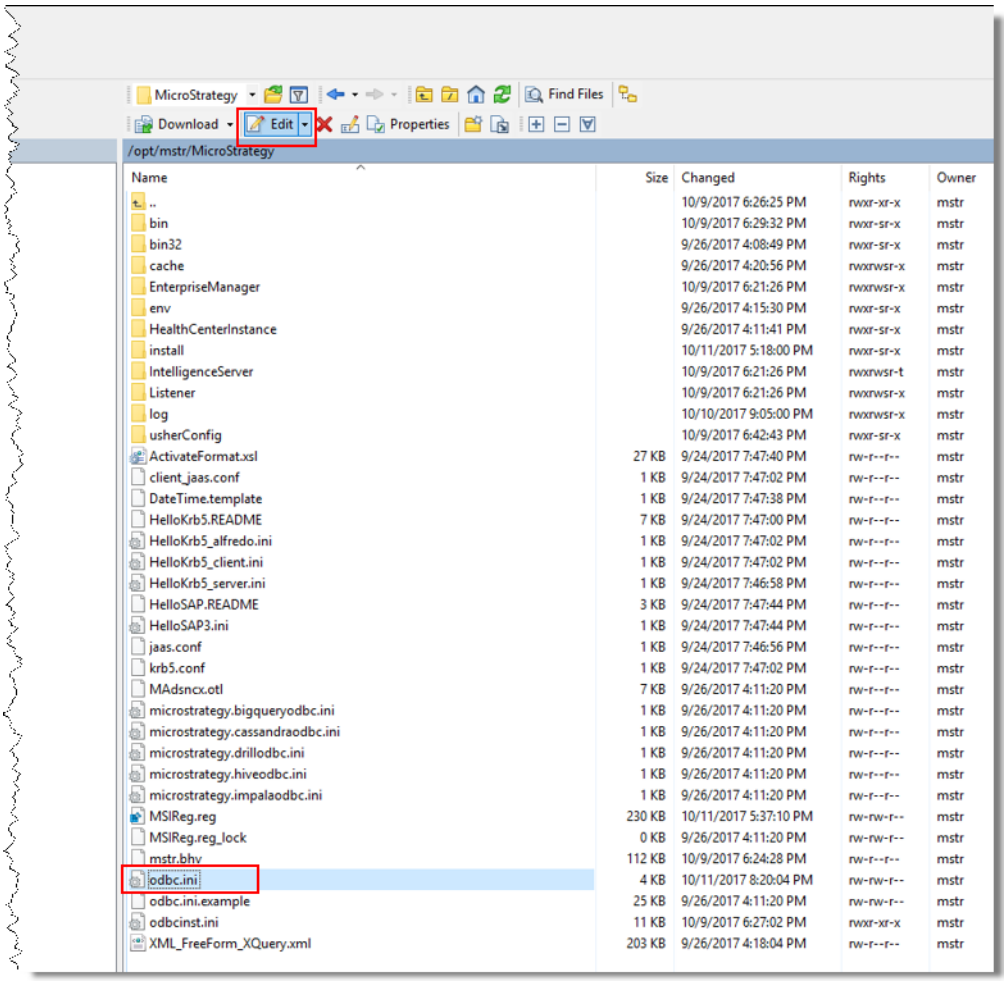
3 In the **Password** box, type the password from your MicroStrategy Cloud email.

4 In the WinSCP window, in the drop down list on top of the right pane, select the **/<root>** directory and browse to **\opt\mstr\MicroStrategy**.

5 In the right pane, select **odbc.ini** and click **Download**.

6 Click **OK**. The file is downloaded to the window machines. This file will serve as a backup.

7 In the WinSCP window, in the right pane, select **odbc.ini** and click **Edit**.



odbc.ini displays in the text editor.

8 Scroll to the section that begins with [ODBC].

9 Update the [ODBC] section to the following:

[ODBC]

Trace=**1**

TraceFile=odbctrace.out

TraceDll=/opt/mstr/MicroStrategy/install/lib/MYtrcXX.so

InstallDir=/opt/mstr/MicroStrategy/install

IANAAppCodePage=106

UseCursorLib=0

UNICOD=UTF -8

10 **Save** the file.

11 Open the MicroStrategy Tutorial project and run a few reports.

12 Edit the **odbc.ini** file again and turn off the trace. To do this, modify the Trace=1 line to **Trace=0** and click **Save**.

13 In WinSCP, navigate to **/opt/mstr/MicroStrategy/IntelligenceServer** and find the **odbctrace01.out** odbc trace file.

14 Open the file and review the ODBC connection information.

The ODBC trace consumes system resources whenever a data source connections are used by MicroStrategy. To ensure system resources are readily available to business users, only use the ODBC trace for brief troubleshooting periods.

7

Maintaining the Infrastructure

InfiniRec's infrastructure is running smoothly, and is consistently being monitored for any resource problems. Based on the information gathered during monitoring, you can fine-tune your infrastructure to accommodate the demand placed on your resources.

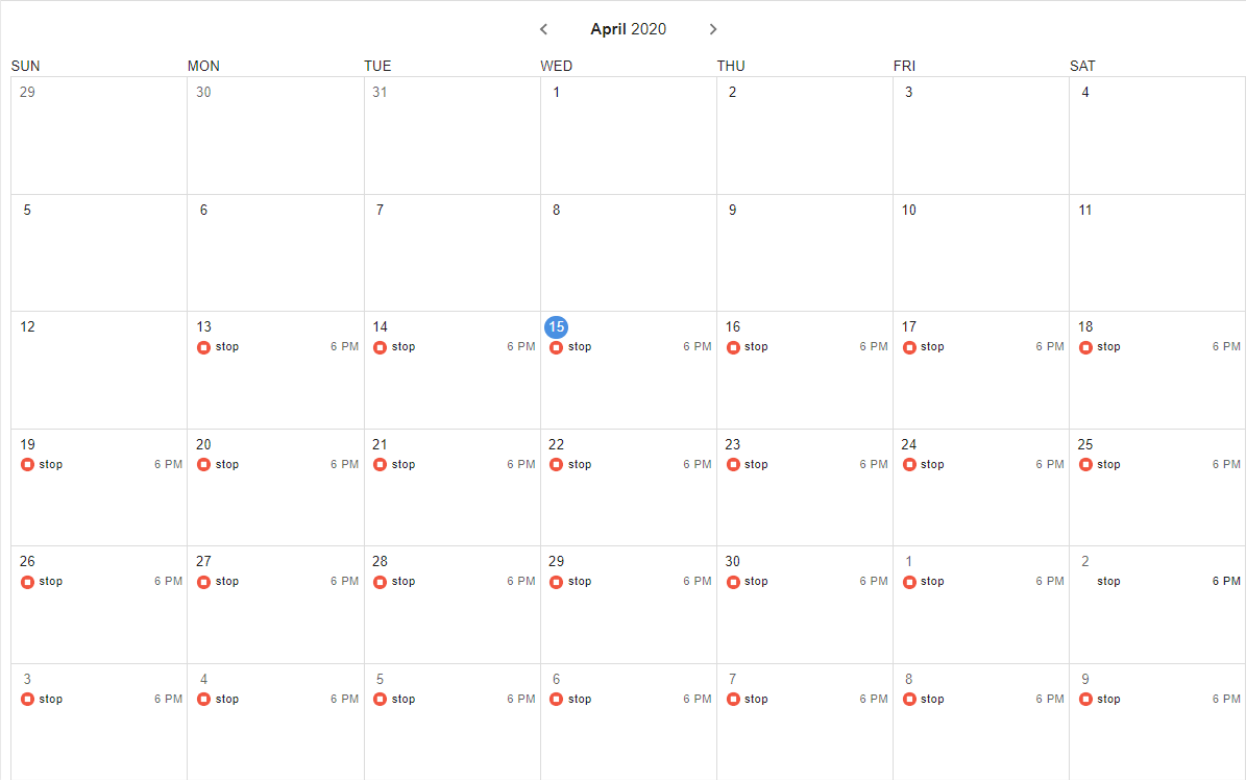
In this chapter, you will learn to create protocols to help administrators reliably perform the following infrastructure maintenance activities:

- **Hardware maintenance**: perform maintenance and update tasks to reduce costs and ensure that the infrastructure is able to meet user demand when required.

Maintaining hardware to ensure efficiency

To keep your infrastructure running efficiently, it is important to maintain its components. An efficient infrastructure enables you to save financial resources by eliminating hardware waste. For example, you might schedule a MicroStrategy

Cloud environment to shutdown every day at 6:00 PM to mitigate waste, as in the following example.



To ensure that your infrastructure performs efficiently, create standards that enable infrastructure administrators to maintain and upgrade hardware components. The standards in your organization might include the following maintenance tasks:

- Perform periodic maintenance tasks and update the operating system to keep it operating efficiently. For example, you can create a schedule to apply security patches, perform malware scans, disable unnecessary startup programs, and install feature updates.
- Perform periodic maintenance tasks and upgrade the network and file system. For example, to ensure that the file system performs optimally, you can set a schedule for tasks such as disk defragmentation, error checking, temporary file deletion, and general disk cleanup.
- Upgrade to the latest certified ODBC drivers and native connectors to ensure that latest features and performance enhancements can be leveraged. For example, when you upgrade the MicroStrategy platform to the latest version, view the Readme to identify the latest drivers and connectors for your data sources.

- Turn off unused machines when not in use and decommission abandoned machines to prevent unnecessary costs. For example, you can turn off unused MicroStrategy Cloud environments during periods of low activity such as holidays or weekends. This practice allows you to save financial resources for periods of peak activity.

Demo 7.1: Create a schedule to start and stop your cloud environment

To cut infrastructure costs and create an efficient budget, you can schedule your MicroStrategy Cloud environments to operate on a predetermined schedule.

For example, your organization may create the following cloud environments to serve distinct purposes:

- A development environment where users create preliminary versions of datasets and install new versions of the MicroStrategy platform.
- A testing environment where development work is moved for extensive testing before production.
- A production environment that is used for live analytical reporting.

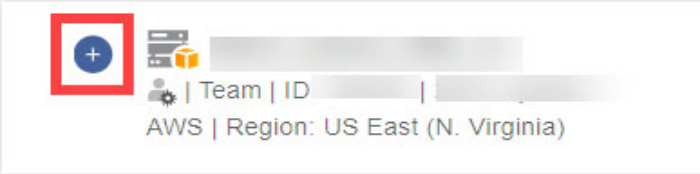
You likely do not need all of your environments to run at all times, and there may be times when some environments can be shut off to save costs. You can leverage a start and stop schedule to manage your environment downtime on a daily, weekly, or monthly schedule.

In this demo, your instructor will create a schedule to keep an environment on between 9AM and 5PM Monday through Friday.

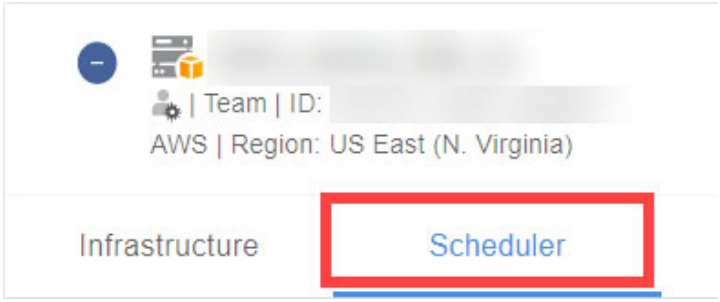
Create a start and stop schedule

1 From your browser, open **https://provision.customer.cloud.microstrategy.com**

2 Click the **+** symbol to the left of an environment.

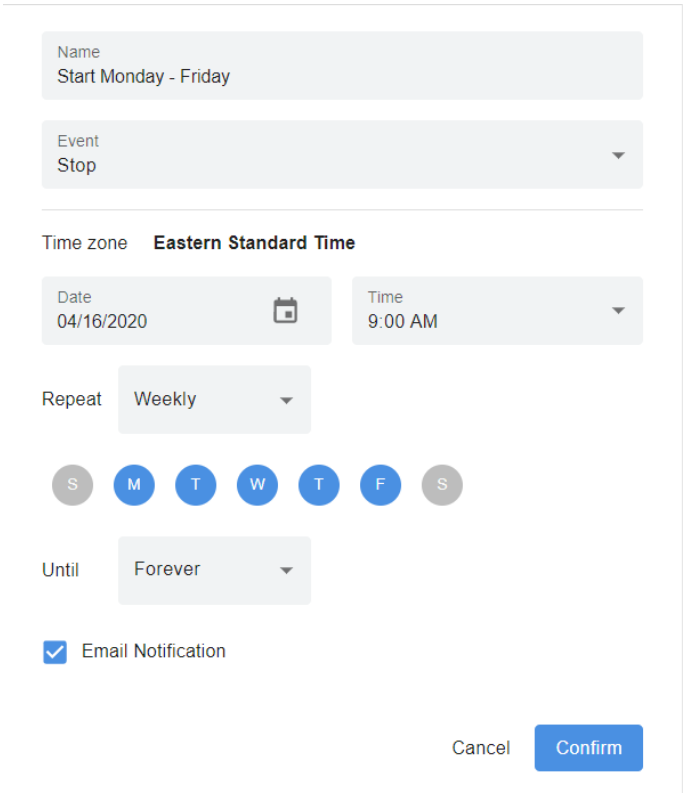


3 Click **Scheduler**.



Create a startup schedule

4 Click **Schedule an Action**, and do the following to create a start schedule:



a In the **Name** box, type **Start Monday - Friday**.

b From the **Event** drop-down list, select **Start**.

c In the **Date** box, click **Calendar** and select the next weekday. The schedule will begin on the selected date.

d From the **Time** drop-down list, select **9:00 AM**.

e From the **Repeat** drop-down list, select **Weekly**.

f Click **M**, **T**, **W**, **T**, and **F** to start the environment during the week.

g From the **Until** drop-down list, select **Forever**. The schedule runs until you delete or modify it.

h Click the **Email Notification** check box to receive an email each time the environment starts according to this schedule.

i Click **Confirm**. The start schedule is added to the calendar.

Create a shutdown schedule

5 Click **Schedule an Action** and do the following to create a Stop schedule:

a In the **Name** box, type **Stop Monday - Friday**.

b From the **Event** drop-down list, select **Stop**.

c In the **Date** box, click **Calendar** and select the next weekday. The schedule will begin on the selected date.

d From the **Time** drop-down list, select **5:00 PM**.

e From the **Repeat** drop-down list, select **Weekly**.

f Click **M**, **T**, **W**, **T**, and **F** to apply the schedule during the week.

g From the **Until** drop-down list, select **Forever**. The schedule runs until you delete or modify it.

h Click the **Email Notification** check box to receive an email each time the environment is stopped according to this schedule.

i Click **Confirm**. The stop schedule is added to the calendar.

Demo 7.2: Creating a schedule to modify environment resources

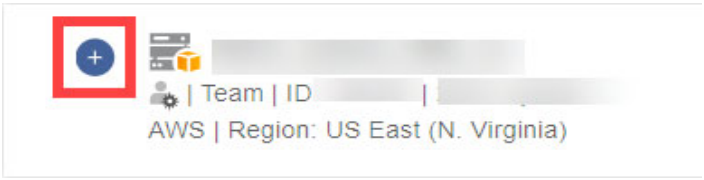
You can modify the hardware resources in your MicroStrategy Cloud environments to accommodate fluctuations in utilization during certain time periods. For example, InfiniRec's corporate office heavily utilizes a specific environment on Monday mornings as corporate analysts come in and begin to review weekend sales data. To ensure that the environment meets performance expectations during this period, you would likely resize the environment for a specific period of time.

In this demo, your instructor will create a schedule that scales an environment up between 8AM and 6PM on Mondays to address an expected weekly utilization spike.

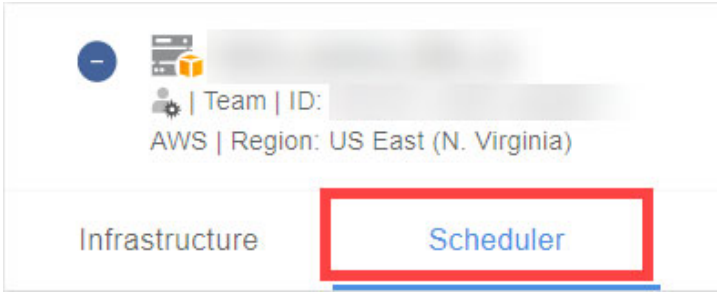
Create a schedule to resize your MicroStrategy Cloud environment

1 From your browser, open **https://provision.customer.cloud.microstrategy.com**

2 Click the + symbol to the left of the environment you created.

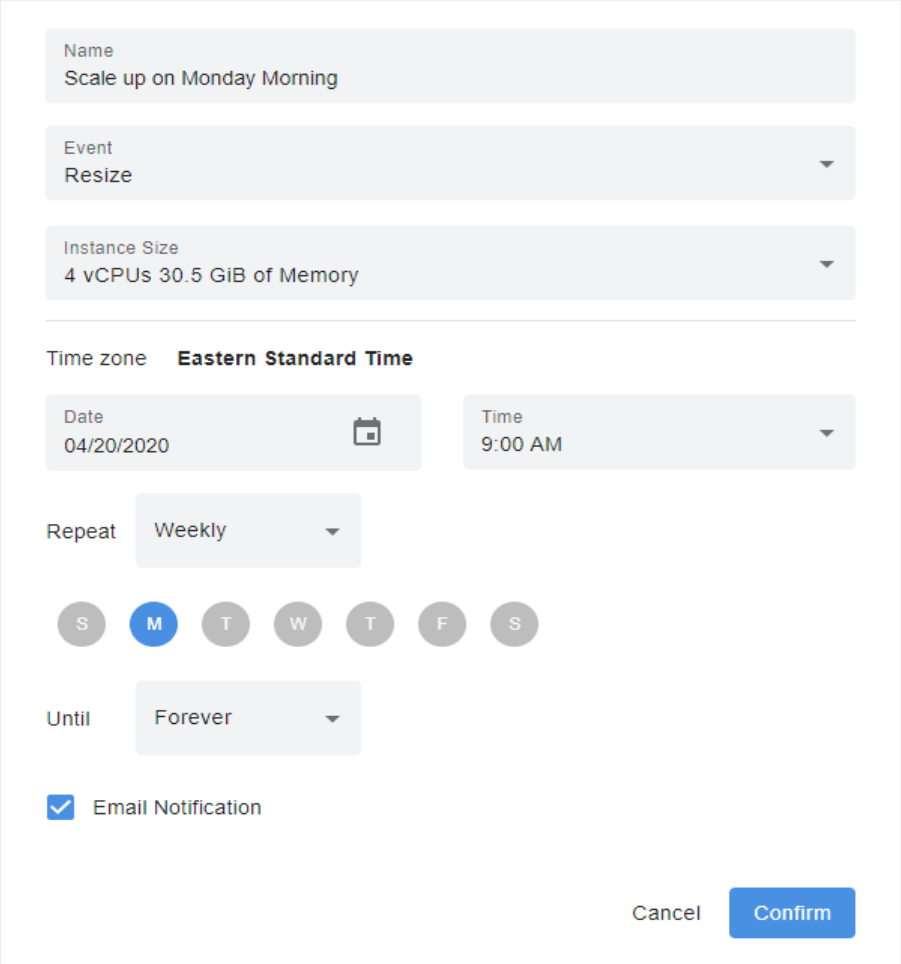


3 Click **Scheduler**.



Create a schedule to scale up the environment

4 Click **Schedule an Action**, and do the following to create a schedule that increases resources.



a In the **Name** box, type **Scale Up on Monday Morning**.

b From the **Event** drop-down list, select **Resize**.

c From the **Instance Size** drop-down list, select **4 vCPUs 30.5 GiB of Memory**.

d In the **Date** box, click **Calendar** and select the next Monday. The schedule will begin on the selected date.

e From the **Time** drop-down list, select **8:00 AM**.

f From the **Repeat** drop-down list, select **Weekly**.

g Click **M** to apply the schedule to Mondays.

h From the **Until** drop-down list, select **Forever**. The schedule will apply until you delete or modify it.

i Click the **Email Notification** check box to receive an email each time the schedule is triggered.

j Click **Confirm**. The schedule is added to the calendar.

Create a schedule to scale down the environment

5 Click **Schedule an Action**, and do the following to create a schedule that resizes hardware resources back to their original level.

a In the **Name** box, type **Scale Down on Monday Evening**.

b From the **Event** drop-down list, select **Resize**.

c From the **Instance Size** drop-down list, select **2 vCPUs 15.5 GiB of Memory**.

d In the **Date** box, click **Calendar** and select the next Monday. The schedule will begin on the selected date.

e From the **Time** drop-down list, select **6:00 PM**.

f From the **Repeat** drop-down list, select **Weekly**.

g Click **M** to apply the schedule to Mondays.

h From the **Until** drop-down list, select **Forever**. The schedule will apply until you delete or modify it.

i Click the **Email Notification** check box to receive an email each time the schedule is triggered.

j Click **Confirm**. The schedule is added to the calendar.

You created schedules to scale the environment resources to address a regular weekly spike in utilization.

A

Appendix: System Administrator Checklist

System Administrator description

The System administrator is responsible for the Enterprise Intelligence compute environments (either on-premise, Cloud-based, or hybrid), including hardware, operating systems, network, file systems, database and data source connectivity, authentication providers, firewalls, application server configuration, monitoring and alerting tools, backup and recovery strategies, and failover strategies.

The Enterprise Intelligence compute environment is the compute environment dedicated to hosting the Enterprise Intelligence platform and related tools.

Task and tools check list summary

Assess

1. Environment uptime
2. Server uptime
3. Services uptime
4. Average CPU
5. Maximum CPU
6. Average system memory
7. Maximum system memory
8. Average network efficiency
9. Free disk space
10. Operating cost

Plan

1. Operating system lifecycle management
2. Hardware lifecycle management
3. Network architecture
4. Drivers lifecycle management
5. Server monitoring and alerting tools lifecycle management
6. Environment fault tolerance architecture
7. Network security architecture
8. Directory services
9. Application server configuration components

Create

1. Server file system
2. Database and data source connectivity
3. Server alerts and notifications

Publish

1. Intelligence compute environment
2. Standard operating procedures
3. Environment SLAs
4. Server log data

Operate

1. Monitor compute environment reports
2. Support Platform Administrator
3. Support Intelligence Center Architects
4. Handle compute environment cases
5. Troubleshoot compute environment issues
6. Coordinate with Intelligence Center

Optimize

1. Environment performance
2. Environment resource requirements
3. Environment reliability
4. Environment efficiency
5. Environment fault tolerance

Assets and tooling

General

- MicroStrategy product documentation
- Operating system documentation
- Hardware, network, and file system documentation
- MicroStrategy Developer
- MicroStrategy Intelligence Server

Assess

- Heartbeat Monitor (or a similar tool)
- Pingdom (or a similar tool)
- MicroStrategy Dossier
- New Relic (or a similar tool)
- Cloudwatch (or a similar tool)

Plan

- MicroStrategy product documentation
- Operating system documentation
- ODBC drivers
- Native connectors
- Heartbeat Monitor (or a similar tool)
- New Relic (or a similar tool)
- Directory services documentation
- Hardware, network and file system documentation

Create

- MicroStrategy Connectivity Wizard

- odbc.ini, odbc.ini.example, ODBC.sh
- MicroStrategy download site
- Microsoft ODBC Administrator

Publish

- Draw.io (or similar)
- Operating system logs
- Network logs

Operate

- Heartbeat Monitor (or a similar tool)
- Linux commands/scripts
- MicroStrategy Developer
- ODBC trace
- Core files
- pstacks (Linux)
- Hang mode dumps (Windows)
- Performance Monitor (Windows)
- New Relic (Linux, Windows)

Optimize

- MicroStrategy Intelligence Server
- Configuration settings
- MicroStrategy query engine backend settings
- MicroStrategy Developer
- Performance Monitor (Windows)
- New Relic (Linux, Windows)

Detailed check list

Uptime

Key Performance Indicators

- Environment uptime
- Server uptime
- Services uptime

Troubleshooting

- Initiate procedures to perform a root cause analysis, if any of the uptimes are less than100%,
- Begin immediate troubleshooting to determine a root cause if any of the uptimes fall below 95%, and present actionable items for system remediation within 2 hours.

Usage

Key Performance Indicators

- Average CPU
- Maximum CPU
- Average system memory
- Maximum system memory
- Average network efficiency
- Free disk space

Troubleshooting

- Consider the following optimization strategies, if the average CPU is greater than a predefined threshold (for example, 80%):
  - Procuring a faster processor

    - Increasing the number of processing units

- Consider increasing memory on the Intelligence Server machine if the average memory usage on that machine is greater than a predefined threshold (for example, 80%)
- Consider the following strategies for optimizing network usage if the average network efficiency (measured by network throughput as a percentage of network bandwidth) is consistently greater than a predefined threshold (for example, 60%):
    - Increase network bandwidth
    - Decrease proximity between the various machines (Intelligence Server, Web/Mobile Server, metadata, and warehouse)
    - Implement server administration solutions such as additional element/dataset caching and incremental fetching of result sets

Cost

Key Performance Indicator

- Operating cost

Troubleshooting

- Work with the Intelligence Center team to discuss strategies for cost optimization, if the operating cost exceeds a predefined threshold.

Plan

Operating System lifecycle management

- Determine the type of Enterprise Intelligence compute environment (on-premise, Cloud, or hybrid)
- Plan operating systems for hosting MicroStrategy servers and MicroStrategy clients
    - Decide the type of operating system for each MicroStrategy server and client installation
    - Determine the latest version of the operating system to be installed

- Install operating systems for hosting MicroStrategy servers and clients
  - Install the certified version of the operating system
  - Install the recommended filesets, security patches and libraries
- Apply latest patches and security updates to the operating system
- Create a service account for the Platform Administrator to:
  - Install and register MicroStrategy Intelligence Server
  - Install and register MicroStrategy Web and MicroStrategy Mobile Servers
  - Install MicroStrategy client products and tools
- Provide the service account created for the Platform Administrator with appropriate permissions to:
  - Install or update system level files during the installation
  - Register and run Intelligence Server as a service
  - Access the Home, Log, Inbox, Cube, and Cache storage directories
- Create a service account for the Database Architect to install database servers

Hardware lifecycle management

- Provision hardware for the MicroStrategy Intelligence Server machine
  - Perform system sizing and capacity planning to provision the processor type, processor speed and number of processing units (processors or cores) to be deployed for MicroStrategy Intelligence Server usage
  - Determine if Hyper-Threading/Simultaneously Multi-Threading (SMT) needs to be enabled for MicroStrategy Intelligence Server performance
  - Perform system sizing and capacity planning to provision memory (physical and swap memory) for the MicroStrategy Intelligence Server machine
  - Perform system sizing and capacity planning to provision storage (hard disk space) for the MicroStrategy Intelligence Server machine
- Provision hardware for MicroStrategy Web and Mobile Server machines
  - Provision the processor type, processor speed and the number of processing units (processors or cores) to be deployed for MicroStrategy

Web and Mobile Server usage, as per recommendations in the MicroStrategy product documentation

  - Provision memory (physical and swap memory) for the MicroStrategy Web and Mobile Server machines, as per recommendations in the MicroStrategy product documentation
  - Provision storage (hard disk space) for the MicroStrategy Web and Mobile Server machines, as per recommendations in the MicroStrategy product documentation

- Provision Hardware for MicroStrategy client machines
  - Provision the processor type for client machines on which MicroStrategy products and tools will be installed, as per recommendations in MicroStrategy product documentation
  - Provision memory for client machines on which MicroStrategy products and tools will be installed, as per recommendations in MicroStrategy product documentation
  - Provision storage (hard disk space) for client machines on which MicroStrategy products and tools will be installed, as per recommendations in MicroStrategy product documentation
- Provision additional storage (hard disk space) for MicroStrategy Server and client installations
  - Factor in storage for installation of MicroStrategy common files
  - Allocate free space in temporary directory for Linux, AIX and Solaris
  - Plan adequate storage space for Cache, History and Cube files
  - Provision hardware for database server machines
- Work with the Database Architect to provision the processor type, processor speed, number of processors, memory, and storage (hard disk space) for database server machines

Network Architecture

- Provision a shared and secure network location where MicroStrategy installation files will be made available
- Determine the physical location of database servers, MicroStrategy Intelligence Server, MicroStrategy Web Server, MicroStrategy Mobile Server and MicroStrategy client machines, and estimate the amount of bandwidth between them

- Register the Server and client machines in the Domain Name Server (DNS)

Drivers lifecycle management

- Install certified ODBC drivers not shipped with MicroStrategy products
  - Install dependent files and libraries
- Update ODBC drivers
- Install native data source connectors
  - Install dependent files and libraries
- Update native data source connectors

Server monitoring and alerting tools lifecycle management

- Install monitoring tools (such as Heartbeat Monitor) to monitor key parameters of the servers, services, operating systems, and network: CPU usage, memory usage, I/O requests, free disk space, file descriptor count (Linux), network usage and network bandwidth

Environment Fault Tolerance Architecture

- Plan schedules to backup Caches, Cubes and History List files
- Plan schedules to backup ODBC configuration files such as odbc.ini and ODBC.sh
- Plan schedules to backup MicroStrategy configuration, application and customization files
- Create contingency and failover plans to handle emergencies:
  - Operating System crash
  - Network outage
  - Hardware failure
  - Database failure

Network Security Architecture

- Secure communications between server and client components (Tunneling, VPN, SSL/TLS, Certificates, Proxies)
- Determine ports that need to be used for communication and enable them

Directory Services

- Work with the Platform Administrator to determine the mode of authentication for the MicroStrategy environment
- Plan for procurement and access to the third-party authentication server/tool (LDAP, Kerberos, Siteminder, Tivoli, Ping Federate, Oracle Access Manager, SAML)
- Work with the Platform Administrator to setup connectivity to the third-party authentication server/tool

Application Server Configuration Components

- Install and configure a certified version of the Web Application Server and Mobile Application Server, as per recommendations in MicroStrategy product documentation
- Install and configure a certified version of the Portal Server, as per recommendations in MicroStrategy product documentation
- Work with Platform Administrator to configure integration with third party mapping services (ArcGIS, ESRI or Google maps)
- Setup clustering/load balancing for MicroStrategy Web and Mobile Server
- Apply latest patches and updates to Application Server Configuration components

Create

Server File System

- Download MicroStrategy product installation files at an accessible location
- Provide the Platform Administrator access to this location to allow for installation of MicroStrategy software

- Create location for Cache, History and Cube files
  - Map/mount location for service account to access these files
- Enable collection of core files and provide location for it, as these files may be needed to troubleshoot MicroStrategy Intelligence Server stability (crash) issues

Database and Data Source Connectivity

- Configure ODBC connectivity for MicroStrategy-shipped drivers using MicroStrategy Connectivity Wizard
- Configure ODBC connectivity for drivers not shipped with the MicroStrategy product
  - Use MicroStrategy Connectivity Wizard or Microsoft ODBC Administrator on Windows
  - Use odbc.ini, odbcinst.ini and ODBC.sh files on Linux
- Update ODBC connectivity information in response to configuration changes
  - Changes in database server name or IP address, port number, or other configuration should be reflected in the ODBC connectivity parameters
- Configure native connectivity to Big Data and related data sources
- Update native connectivity information in response to configuration changes
  - Changes in data source server name or IP address, port number, or other configuration should be reflected in the data source connectivity parameters

Server Alerts and Notifications

- Set up alerts to notify Platform and System Administrators when key parameters exceed predefined threshold limits

Publish

Intelligence Compute Environment

- Publish the topology diagram of the Enterprise Intelligence compute environment using a flowchart builder tool such as Draw.io

    - The topology diagram comprises a detailed map showing all components of the Enterprise Intelligence compute environment, including cloud and on-premise environments
- Make this diagram available to the Intelligence Center

Standard Operating Procedures

- Document operating procedures for all components of the Enterprise Intelligence compute environment, including the operating system, network, and file system
- Publish maintenance window schedule for operating system, hardware, and other components of the Enterprise Intelligence compute environment
    - Notify the Intelligence Center in advance so that the team can be prepared for the maintenance window and the Platform Administrator can notify end users

Environment SLAs

- Document and publish SLAs for the Enterprise Intelligence compute environment

Server Log Data

- Enable and maintain logging that tracks activity of the operating system, network, and other components of the Enterprise Intelligence compute environment
    - Furnish these logs for analysis and troubleshooting when required
- Enable additional or detailed logging/diagnostics for the Enterprise Intelligence compute environment, when required for troubleshooting

Operate

Monitor Compute Environment Reports

- Monitor CPU usage, memory usage, and I/O requests of Intelligence Server, Web and Mobile Server processes, using Heartbeat Monitor (or a similar tool), to proactively prevent these servers from crossing predefined thresholds

- Monitor the free disk space on the operating system using New Relic (or a similar tool) to proactively prevent system crashes due to lack of disk space
- Monitor the count of file descriptors for the Intelligence Server process on Linux using commands/scripts. Consistent increases in the number of file descriptors may be symptomatic of a larger problem.
- Monitor network efficiency using New Relic (or a similar tool) to ensure that the network is not getting overloaded
- Monitor uptimes for the following using New Relic (or a similar tool):
    - Environment (production, test, development)
    - Servers (Intelligence Server, Web Server, Mobile Server)
    - Services (MicroStrategy Listener, MicroStrategy Enterprise Manager Data Loader, Directory)
- Monitor physical disk utilization on the Intelligence Server machine using New Relic (or a similar tool) to ensure that the operating system is not frequently swapping memory between the physical RAM and disk

Support Platform Administrator

- Perform the following actions immediately if MicroStrategy Intelligence Server has an abnormal shutdown (crash):
    - Work with the Platform Administrator to bring the Intelligence Server back up and running as quickly as possible
    - Upload the core files to a location where they can be accessed by the Platform Administrator and sent to MicroStrategy Technical Support for root cause analysis
    - Install any patches issued by MicroStrategy Technical Support to resolve Intelligence Server crash issues
- Perform the following actions immediately if MicroStrategy Intelligence Server becomes unresponsive (hangs):
    - Work with the Platform Administrator to bring the Intelligence Server back up and running as quickly as possible
    - Collect pstacks (Linux) or hang mode dumps (Windows)
    - Upload the pstacks/dumps to a location where they can be accessed by the Platform Administrator and sent to MicroStrategy Technical Support for root cause analysis

    - Install any patches issued by MicroStrategy Technical Support to resolve Intelligence Server hang issues

- Perform the following actions immediately if MicroStrategy Intelligence Server shuts down due to memory depletion:
    - Work with the Platform Administrator to implement memory contract management governors within the MicroStrategy product to prevent the Intelligence Server from shutting down
    - Facilitate collection of operating system performance logs:
        - Performance Monitor logs (Windows)
        - New Relic (Linux, Windows)
    - Upload the performance logs to a location where they can be accessed by the Platform Administrator and sent to MicroStrategy Technical Support for root cause analysis
    - Install any patches issued by MicroStrategy Technical Support to resolve Intelligence Server memory depletion issues

Support Intelligence Center Architects

- Coordinate with Intelligence Center Architects to communicate urgent issues or areas of improvement pertaining to the Enterprise Intelligence compute environment using a collaboration tool

Troubleshoot Compute Environment Issues

- Troubleshoot issues pertaining to the operating system, network, file system, hardware, directory services and drivers
    - Provide resolution within timelines that meets SLA requirements
    - Work with the vendor/provider directly to achieve a resolution
- Enable and facilitate ODBC trace logs to troubleshoot issues regarding connectivity to, query execution in, and data fetching from the database servers

Coordinate with Intelligence Center

- Attend daily Intelligence Center scrum meeting

    - Provide update on the status of the Enterprise Intelligence compute environment

Optimize

Environment Performance

- Work with the Platform Administrator to tune Intelligence Server configuration settings, to minimize file descriptor usage on Linux Operating System for the Intelligence Server process:
    - Optimize number of database connections
    - Optimize number of network connections
    - Use effective Cache, Cube and History List file management strategy – balance the number and frequency of file access versus report/document/dossier performance
    - Minimize the number of log (diagnostic) files
- Optimize network usage, if the network efficiency (network throughput as a percent of network bandwidth) is consistently greater than a predefined threshold (for example, 60%), as this may be indicative of the network negatively affecting system performance:
    - Increase network bandwidth
    - Work with Platform Administrator to consider alternative strategies within MicroStrategy server administration to alleviate high network usage:
        - Additional element and dataset caching
        - Incremental fetching of result sets
    - Consider placing all server components in the same network segment to reduce bottlenecks and increase bandwidth
    - Consider lessening distance between Intelligence Server and the metadata and warehouse to optimize network usage
    - Consider tuning backend settings in the MicroStrategy product to optimize data chunking and data retrieval across the network

Environment Resource Requirements

- Work with the Platform Administrator to determine if processor speed and/or number of processing units needs to be increased
    - Consider procuring a faster processor, or increasing the number of processing units, if the processor is consistently running at high capacity (greater than a predefined threshold such as 80%)
- Work with the Platform Administrator to determine if the MicroStrategy Intelligence Server machine memory needs to be increased
    - Consider increasing memory on the Intelligence Server machine under the following circumstances:
        - Memory counters indicate that Intelligence Server is consistently using more than a predefined threshold of the Operating System memory resources (for example 80%)
        - Intelligence Server has significant memory contract request rejections
        - Physical disk utilization counters consistently exceed a predefined threshold (for example 80%)

Environment Reliability

- Conduct a comprehensive operational review of the operating system, network and file system
    - Use Performance Monitor logs (Windows) and New Relic (Linux, Windows) to analyze operating system, network and file system statistics periodically and verify that key parameters (CPU usage, memory usage, I/O requests, free disk space, file descriptor count) are consistently operating within set performance benchmarks
- Perform root cause analysis on any significant violation of these benchmarks

Environment Efficiency

- Perform periodic maintenance and upgrades to the operating system to keep it operating efficiently
- Perform periodic maintenance and upgrades to the network and file system
- Upgrade to latest certified ODBC drivers and native connectors
- Decommission machines that are no longer in use to prevent unnecessary costs

Environment Fault Tolerance

- Optimize resources across test, development and production environments
  - Utilize resources from test and development environments to temporarily cover production environment in case of failover

Definitions

Term	Definition
Application Server Configuration	Component Server components that facilitate storage, processing and delivery of content across client environments over the web
Authentication Provider	Systems that provide a mechanism to identify and validate users
Backup and Recovery	Strategy defining policies for backing up and recovery of data in case of a loss or a critical event
Capacity Planning	Process of determining the resources needed by each component of the Enterprise Intelligence compute environment to handle current and future workload requirements
Cloud	Software installed and run on shared pools of configurable resources that are rapidly provisioned with minimal management effort
Cloudwatch	Monitoring tool for Cloud environment
Collaboration Tool	Platform (such as Slack) used to communicate information, updates, and ideas between members of the Intelligence Center team
Driver	Tool that facilitates connectivity and access to databases and data sources
Directory Services	Shared information infrastructure for locating, managing, administering, and organizing users, groups, files, printers, and other resources
Environment Fault Tolerance Architecture	Framework to enable the system to continue operating properly in the event of the failure of one or more of its components
Enterprise Intelligence compute environment	Compute environment dedicated to hosting the Enterprise Intelligence Platform and related tools

Term	Definition
Enterprise Intelligence Platform	Set of methodologies, processes, and architectures that transform raw data into meaningful information to enable strategic operational insights and decision-making
Environment Efficiency	Effectiveness of delivery in relation to planned capacity for each component of the Enterprise Intelligence compute environment
Environment Performance	Responsiveness of each component of the Enterprise Intelligence compute environment within set benchmarks and guidelines
Environment Reliability	Availability and robustness of each component of the Enterprise Intelligence compute environment at all times
Environment Resource Requirements	Resource capacity to be delivered by each component of the Enterprise Intelligence compute environment
Environment SLAs	Agreement between the Enterprise Intelligence compute environment provider and its internal or external customers that documents what services the provider will furnish and defines the performance standards the provider is obligated to meet
Failover	Mechanism for switching to a standby server, hardware component, or network when the primary component becomes unavailable due to failure or scheduled downtime
File System	Platform for storage, organization, and retrieval of file network security system (software-or hardware-based) that uses rules to prevent unauthorized access to or from a network
Hardware	Physical components of a computer or other electronic system
Heartbeat Monitor	Monitoring tool for Linux operating system
Hyper-Threading	Proprietary technology by Intel for parallel computation that allows a processor to execute multiple processes or threads concurrently as supported by the operating system

Term	Definition
Intelligence Center	Team responsible for building and operating an Intelligence Architecture by following the Intelligence programs best practices, to publish Enterprise Applications and Enterprise Mobility Applications to the organization constituents
Kerberos	Network authentication protocol that provides integrated authentication for client/server applications using cryptography
LDAP	Lightweight Directory Access Protocol represents an open standard Internet protocol for applications to request and manage user and group directory information
MicroStrategy Client	Includes products such as MicroStrategy Developer, MicroStrategy Command Manager, MicroStrategy System Manager, MicroStrategy Object Manager, MicroStrategy Integrity Manager
MicroStrategy Server	Includes products such as MicroStrategy Intelligence Server, MicroStrategy Web Server and MicroStrategy Mobile Server
Monitoring and Alerting Tool	Mechanism to continually check the values of key system parameters and notify an administrator when the system is sliding towards undesirable operating conditions
Network Architecture	Framework for the specification of a network's physical components and their functional organization and configuration
Network Security Architecture	Collective measures designed to protect the usability and integrity of the network from illegitimate use, malicious threats and attacks
New Relic	Tool that monitors key parameters of the operating system, network, and file system
odbc.ini	File that provides and maintains ODBC connectivity information to Database Management Systems (DBMS)
odbcinst.ini	File that provides and maintains ODBC connectivity information to Database Management Systems (DBMS) for connections without a defined Data Source Name (DSN)

Term	Definition
ODBC.sh	File that contains location information about ODBC drivers installed on the Operating System
ODBC Connectivity	Standard application programming interface (API) for communicating with Database Management Systems (DBMS)
On Premise	Software installed and run on computers within the premises of the organization
Operating System	Software that manages computer hardware and software resources provisioning common services for computer programs
Pingdom	Tool that monitors whether servers and applications are up and running
Server Log Data	Files automatically created and maintained by a server detailing the list of performed activities
Simultaneous Multi-Threading	Technology for parallel computation that allows a CPU or core to execute multiple processes or threads concurrently as supported by the operating system
Standard Operating Procedures	Document that details instructions for completing tasks and activities associated with regular operations of each component of the Enterprise Intelligence compute environment
System Sizing	Activity that estimates the type, speed, and quantity of the hardware needed for each component of the Enterprise Intelligence compute environment

Copyright Information

Trademark Information