Data Center Discovery 2021.2

Scan Engine Deployment Guide

Rev 06/20

Contents

Introduction

The scanning of large estates presents a complex task, which is compounded when multiple layers of clustering and virtualization are present within the estate. This document helps you make appropriate choices in identifying a suitable scan-engine deployment scenario to meet your company's needs.

The most common deployment scenarios are explained here—if a scenario isn't covered, contact Ivanti support for assistance.

Overview

There are multiple ways to deploy the Ivanti Data Center Discovery scan engine. Ivanti works with each client to decide which type of deployment best suits the needs of each company. The following sections explain the three most common deployment types, accompanied with an example scenario and any special requirements for each scenario:

- Centralized management with a single scan engine
- Centralized management with multiple scan engines
- Centralized management with distributed scanning

See the *Data Center Discovery—Scan Engine Prerequisites Guide* for more information regarding estate requirements for installing the scan engine.

Deployment scenarios

The following scenarios are the most common scan-engine deployments—they do not represent every deployment scenario possible. They're meant as a guide to give you a good idea of how a deployment works. You'll need to work with Ivanti support to decide what type of deployment is best for your company.

You may notice this document provides content similar to the *Data Center Discovery—Scan Engine User Guide* use-case scenarios. The purpose of this document is to identify the possible *installation locations* of the scan-engine components. The purpose of the user guide use cases is to identify the possible *roles* and *responsibilities* within a deployment model.

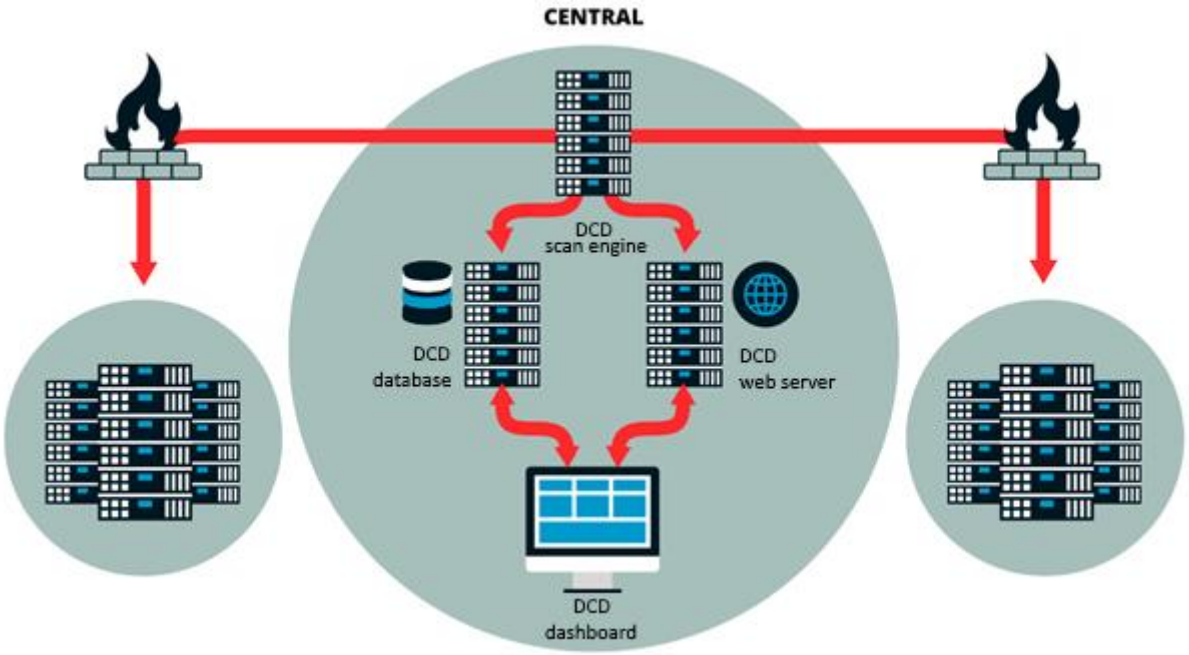
In a typical deployment scenario, the following scan-engine components are deployed:

- **Scanning service**: A service that discovers devices through methods laid out in the "Scan exclusions" section of this document.
- **Scan database**: A joint database that contains:
    - The user interface database—Receives user information from the UI and shares it with the scan database in encrypted form (e.g., credential passwords).
    - Scan data—Stores the raw data from the scan data-retrieval operations.
- **UI**: A web-based user interface used to control and configure the scan operations.
- **REST API**: Programmatic access to the scan-engine scan data. Allows for integration with customer in-house tools.

The scan engine is typically deployed on a single device instance (virtual or physical) within a customer data center.

Centralized with one scanning server

This Data Center Discovery (DCD) deployment is the default for small estates (< 10,000 devices) and proof-of-concept scenarios.



It's the most common type of deployment and requires access to all of the default ports listed in Appendix A and B of this guide. In this scenario, the system administrator and other authorized users centrally manage the scan on a network shared with the servers being scanned. Firewalls between divisions may exist but do not significantly block traffic based on open ports/protocols.

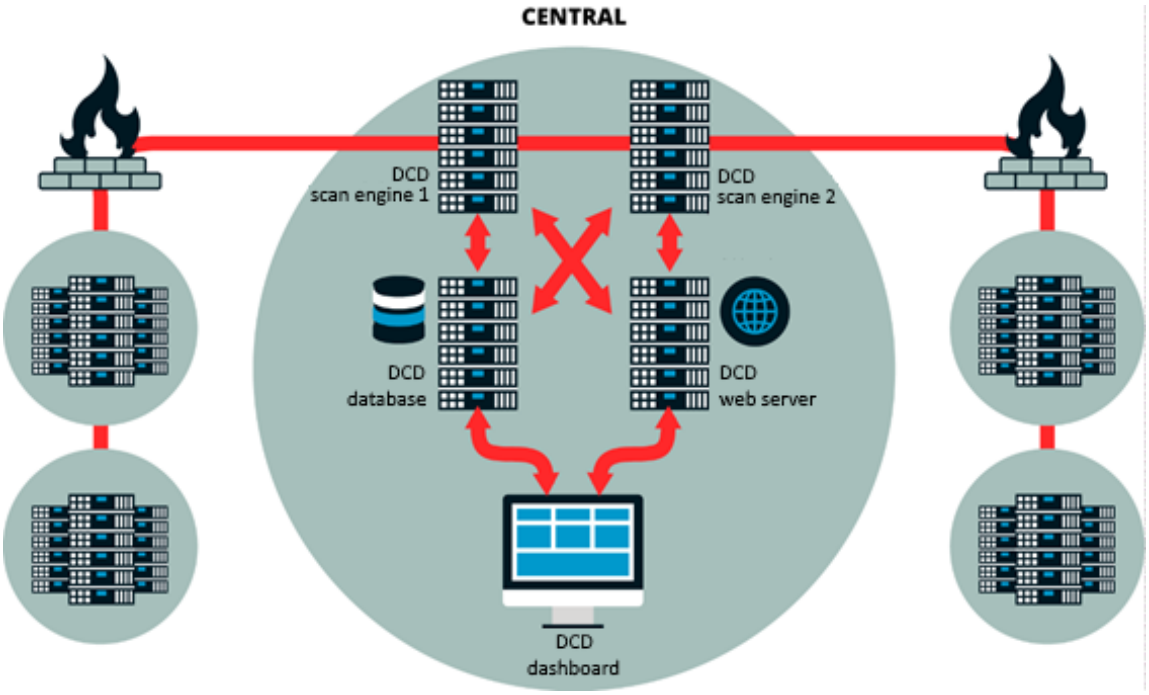
Also, it's important to note that this deployment scenario requires **all scanned traffic** to run through the scan engine to be deposited in the scan-engine scan database. Review the checklist below to see if this deployment best suits your company's needs.

My company:

1. Has **fewer** than 10,000 devices to be scanned.
2. Wants a simple deployment schema.
3. Wants to centrally manage the scan.
4. Will grant access to the default ports listed in Appendices A and B.

Centralized with multiple scanning servers

This scenario is suitable for larger estates (> 10,000 devices) where the scanning load would be excessive for a single scan engine.



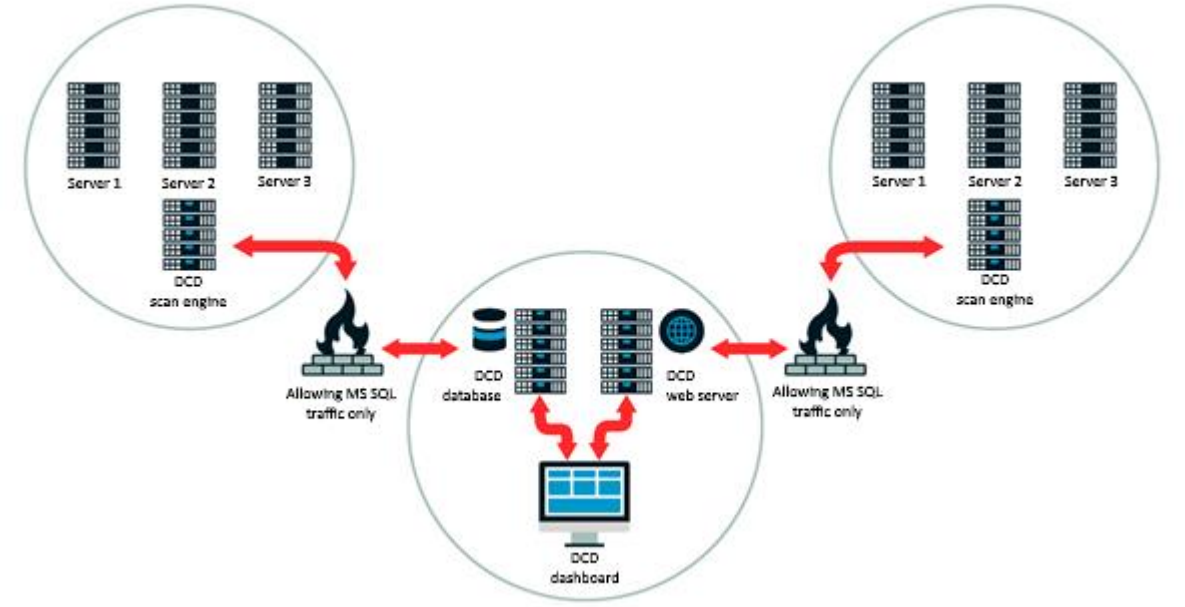
In this scenario, the system administrator and other authorized users centrally manage the scan on a shared network with the servers being scanned. Firewalls between divisions may exist but do not significantly block traffic based on open ports/protocols. This deployment scenario allows **all scanned traffic** to run through the scan engine to be deposited in the shared scan-engine scan database.

My company:

1. Has **more** than 10,000 devices to be scanned.
2. Wants a centrally deployed scan engine.
3. Wants to centrally manage the scan.
4. Will grant access to the default ports listed in Appendices A and B.

Centralized with distributed scanning

This scenario is suitable for both large and small estates where firewalls between business divisions or geographic areas are very restrictive and only allow limited modification to the firewall rules.



This design places the scan engines on the networks that are being scanned. In this scenario, the system administrator and other authorized users centrally manage the scan on a separate network from the servers being scanned. This deployment scenario allows only **SQL traffic** to run through the scan engine to be stored in the scan-engine scan database. Review the checklist below to see if this deployment best suits your company's needs.

My company:

1. Has **more** than 10,000 devices to be scanned.
2. Has restricted traffic flow across firewalls because of security considerations.
3. Wants to centrally manage the scan.
4. Will **not** grant access to the default ports listed in Appendices A and B.

Remote sites with narrow bandwidth

If you have a remote site with narrow bandwidth, you can approach these scans in a way that optimizes performance by doing the following:

- **Enable** scanning on these sites during agreed hours with a reduced thread count and manual supervision of the scan. This scan approach uses significantly fewer bandwidth resources.
- **Install** a remote scanning server in each site that sends its data back to the primary SQL server. This enables control from one location and uses less bandwidth than the first option, as the only information being passed over the WAN connection would be the cleansed results going into the database. Your network engineers can also limit the bandwidth available to this remote scanning server over the WAN.
- **Install** a full scan-engine server on a server OS to laptops or VMs that can be installed in each site. This setup enables the scanning server to scan the site with no communication over the WAN, except for an RDP session to the VM/laptop. The results would have to be managed separately and combined at a later stage.

If needed, you can discuss customer-specific site restrictions with Ivanti support; they'll be happy to assist you.

> **Note:** The definition of **narrow bandwidth** is not specifically defined here, mainly because it's a relationship to the width of the pipe, the number of targets, and the type of scanning that is required (e.g., Windows-based scanning over WMI typically has a higher overhead than SSH-based scanning).

Authentication

Enterprise computer systems typically require some form of authentication before users can begin interacting with them. The creation of credentials associated with the scan operation is typically no more complex that creating a normal login set (either normal username and password or domain-based access).

You can manage credentials for the scan engine in a variety of ways:

- Configure **multiple credential sets**, as well as the order of attempts of the credentials set.
- Configure credential sets for different **device types** (Windows, UNIX, proxies, applications, infrastructure, etc.). Options can be configured for an entire credential set as appropriate.
- Associate an individual **device** or **IP address range** with any credential.
- Configure credential **cool-down algorithms** to control retry attempts or retry intervals and avoid device lock-out.

Any credentials stored within the scan engine are encrypted using RSA encryption with a variable key length. See the *Data Center Discovery—Scan Engine Security Guide* for details about encryption methods.

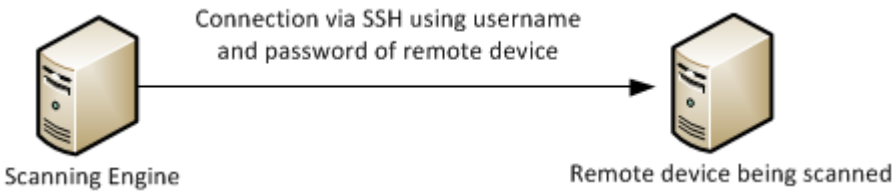
The use of Secure Shell (SSH) has some additional options that are discussed in the following section.

SSH authentication

Enterprise computer systems typically require some form of authentication before users can begin interacting with them. The following sections discuss different mechanisms through which SSH authentication can be supported by the scan engine.

Direct connection

The scan engine establishes a Secure Shell (SSH) connection to a remote device being scanned; when prompted by the remote device, it supplies the configured credentials.



Single sign-on

In medium and large organizations, SSH authentication is accomplished via a Single Sign-On (SSO) mechanism.

The scan engine supports this as follows:

- The scanning server connects via SSH to an SSH proxy using the configured credentials of the SSH proxy.
- The scanning server connects via SSH from the SSH proxy to the remote device to be scanned. The SSH proxy manages the SSO key-based security required by the remote device. (You can configure the command that should be issued to the SSH proxy for the remote device from the scan-engine UI. If required, you can associate a username and password with this command. Configure the command username and password in the UI.)
- Once logged in to the remote device, the scanning server issues the scanning-related commands on the remote device.
- The above process then repeats for each remote device.

You need to consider several **potential constraints** that may be applicable to your organization:

1. Desktop devices are turned off outside office hours.
2. Mobile devices are unavailable for extended periods of time (e.g., annual leave).
3. Dynamic IP addresses or IP leases are for short durations.
4. Different time zones apply (working hours).
5. IP address ranges rarely align with physical locations and overlap is common.
6. Third-party desktop support applies.
7. Credentials are not centrally managed.
8. Limited bandwidth and latency issues from the scanning server to remote offices.

Miscellaneous deployment issues

There are a number of miscellaneous deployment issues that you need to be aware of when planning for a scan.

Locations

Within the scan engine, you can optionally group devices by location. For example:

- Grouping by geography or by data center
- Grouping by DNS locations
- Grouping by business units
- Grouping by service owners

Grouping by locations can be useful for reporting and resolving issues, because the groups break down the scanning problem into manageable sections.

Desktop scanning schedule

Desktop scanning is more time consuming than data-center scanning, especially with a large number of devices and a complex network.

Monitoring of the scanning server

Some administrators may choose to monitor the scanning server to identify any potential issues. The most relevant items to monitor are:

- TCP connection rate
- Bandwidth utilization

Securing your server

Ivanti does not provide guidelines on securing servers used by the scanning server or proxies used in conjunction with the scanning. You can better secure your server by:

- Disabling unused ports. See Appendices A and B of this guide for the default ports lists.
- Ensuring external access to the scanning server is disabled.
- Configuring appropriate levels of IIS logging on the scan-engine server.

Scan exclusions

By default, the scan engine attempts to scan all discovered devices and applications to provide the best overall picture of an estate. Some companies like to exclude certain devices or applications during the scan.

Some reasons for this include:

- Extensive bandwidth constraints
- Shared infrastructure with other companies
- Shared service account with another system or function
- Single gateway
- Outsourced device management
- Need to handle lock-down periods for given devices

Restricting device discovery

If your organization wants to exclude certain items from being scanned, you can add the target IP address to the exclusion list of targets. This is a configuration option on the scan-engine UI.

See the *Data Center Discovery—Scan Engine User Guide* for help with configuring the IP address range(s).

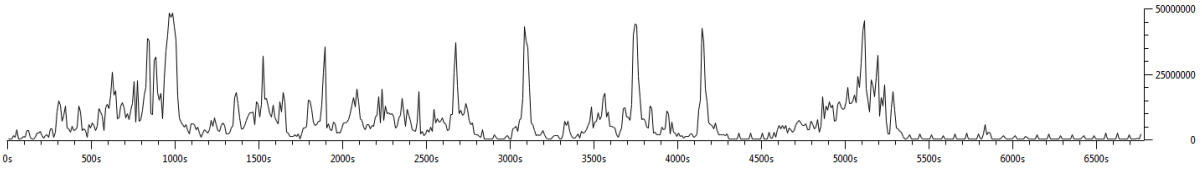
Managing network loads

In general, the amount of network traffic needed can be throttled and controlled.

With sufficient configuration and attention, the amount of network traffic generated for any one physical site within a customer network can be controlled so that slower network locations are inventoried less aggressively, requiring less network bandwidth, while core network segments are aggressively inventoried, decreasing the time taken to audit the bulk of the network.

The scan engine's network utilization and bandwidth requirements depend on the configured services and requirements of any given customer site.

For an unmodified scan-engine installation, expect bandwidth utilization of approximately 3MB/sec. This value is subject to spikes and is not a flat utilization of the network.



The above graph identifies the data flow from the scan engine, with the X-Axis showing timing in seconds and the Y-Axis showing network traffic in bytes per 10-second intervals (peak value of 50,000,000 bytes approximates to 50 MB). This diagram also identifies that the scanning operation is composed of spikes of network traffic and troughs of low bandwidth usage. This is typical of a scanning operation.

The speed at which an inventory runs is a function of network capability, speed of the database server, and scanning server configuration. The faster a scan engine is configured to run, the more bandwidth is required.

Appendix A: Default discovery ports

Port	Function
21	FTP control (command)
22	SSH used for secure logins, file transfers (scp, sftp), and port forwarding
23	Telnet protocol unencrypted text communications
25	Simple Mail Transfer Protocol (SMTP) used for email routing between mail servers
80	Hypertext Transfer Protocol (HTTP)
110	Post Office Protocol 3 (POP3)
135	DCE endpoint resolution
139	NetBIOS Session Service
143	Internet Message Access Protocol (IMAP) used for retrieving, organizing, and synchronizing email messages
443	Hypertext Transfer Protocol over TLS/SSL (HTTPS)
445	Microsoft-DS SMB file sharing
1520	Oracle database common alternative for listener
1521	Oracle database default listener
1522 1529	Oracle database common alternative for listener
3389	Microsoft Terminal Server (RDP) officially registered as Windows Based Terminal (WBT)
7001	Default for BEA WebLogic Servers HTTP server, though often changed during installation

Appendix B: Default inventory ports

Port	Function
22	Secure Shell (SSH) used for secure logins, file transfers (scp, sftp), and port forwarding
23	Telnet protocol unencrypted text communications
80	Hypertext Transfer Protocol (HTTP)
135	DCE endpoint resolution
139	NetBIOS Session Service
443	Hypertext Transfer Protocol over TLS/SSL (HTTPS)
445	Microsoft-DS SMB file sharing
1520-1529	Oracle database Listener
1975	Custom Oracle Database Port
2025	Sybase ASE default 2025
2809	corbaloc:iiop URL per the CORBA 3.0.3 specification
3389	Microsoft Terminal Server (RDP) officially registered as Windows Based Terminal (WBT)
4100	Sybase ASE default 4100
5000	Sybase ASE default 5000
8880	cddbp-alt CD Database (CDDB) protocol (CDDBP) alternate
9043	WebSphere Application Server Administration Console secure
9060	WebSphere Application Server Administration Console
9080	glrpc Groove Collaboration software GLRPC
9088	Informix default port #2
9090	Openfire Administration Console
9100	PDL Data Stream
9402	WebSphere Port
9443	WebSphere Port

Appendix C: Key information

Item	Description
Clustering	When there are multiple devices acting as one.
CMDB	Configuration Management Database
Discovery	The scan engine attempts to find all devices or applications on a network (IP range, hostname, port, etc.).
Discovery source feature	A feature that allows users to restrict scans using database configuration.
Found device	Items (devices) found during a discovery scan.
IIS	Internet Information Services
Inventory scan	When the scan engine scans the items found during the discovery scan.
NETBIOS	Network Basic Input/Output System is a program that allows applications on different computers to communicate within a local area network (LAN).
SSH	Secure Shell (SSH) is a network protocol for secure data communication, remote command-line login, remote command execution, and other secure network services between two networked computers.
Ivanti support	https://www.ivanti.com/support/ivanti-support
Turnover	Devices entering and leaving often on a network.
Virtualization	A virtual version of a device running on a physical host.