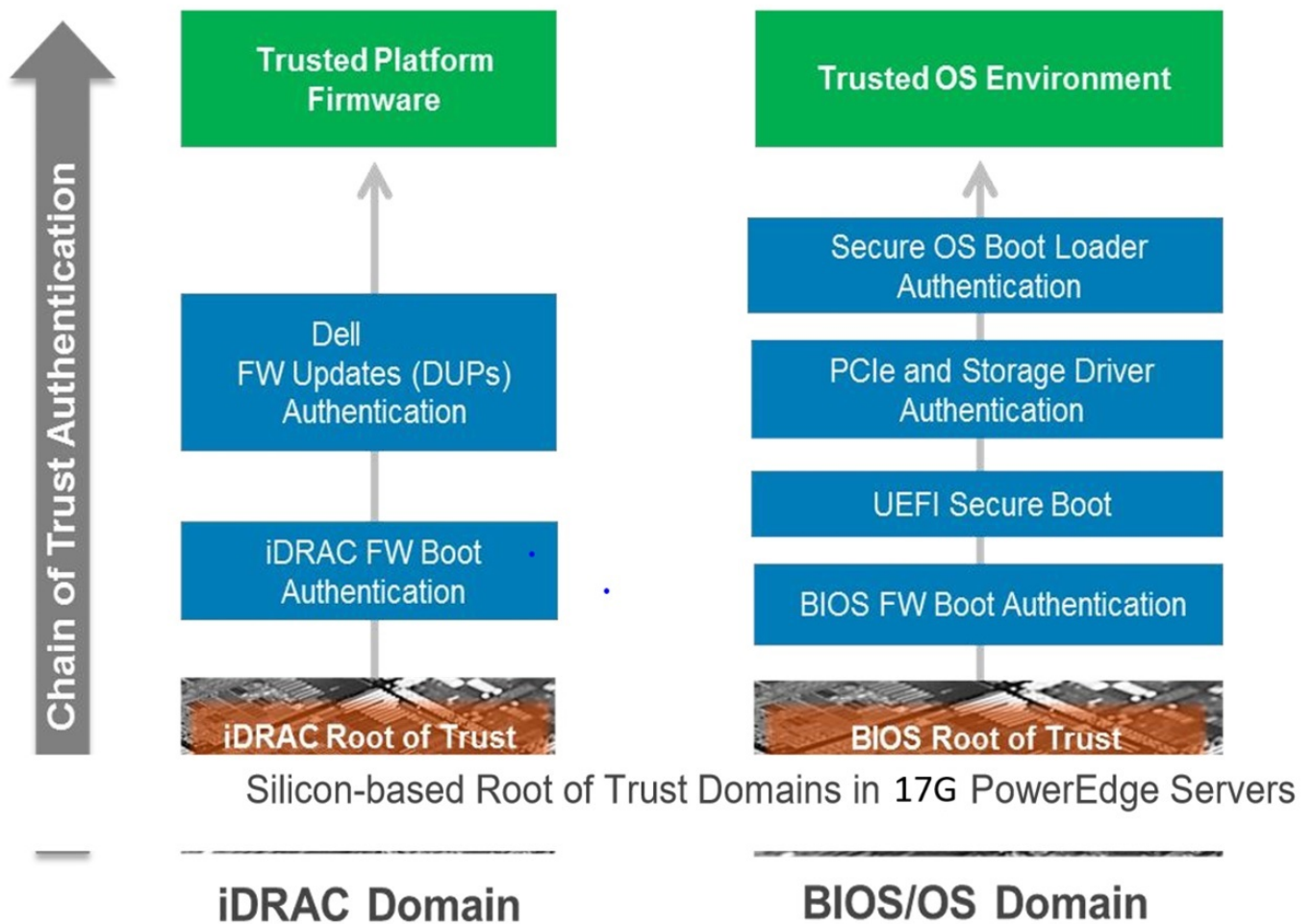


iDRAC10 Version 1.30.xx Security Configuration Guide

1.30.xx Series

Built in iDRAC and PowerEdge Security

The iDRAC boot process uses its own independent silicon-based Root-of-Trust that verifies the iDRAC firmware image. The iDRAC Root-of-Trust also provides a critical trust anchor for authenticating the signatures of Dell firmware update packages (DUPs).



Topics:

- Silicon-based Root-of-Trust
- Cryptographically Verified Trusted Booting
- Signed Firmware Updates
- Non-Root Support
- SELinux
- iDRAC Credential Vault
- BIOS Recovery and Hardware Root of Trust (RoT)
- Live Scanning

Silicon-based Root-of-Trust

PowerEdge servers that are enabled with iDRAC use an immutable, silicon-based Root-of-Trust (RoT) to cryptographically attest to the integrity of BIOS and iDRAC firmware. This Root-of-trust is based on one-time programmable, read-only public keys that provide protection against malware tampering. iDRAC provides enhanced security with the use of a new dedicated security processor with no external interfaces to store all authentication keys and perform iDRAC secure boot. The BIOS boot process leverages Intel Boot Guard technology along with SPDMM measurement verification by iDRAC to verify the digital signature of the cryptographic hash of the boot image. A failure to verify the boot image results in a shutdown of the server and a user notification in the Lifecycle Controller Log. If Boot Guard validates the boot image successfully, the rest of the BIOS modules are validated by using a chain of trust procedure until control is handed off to the operating system or hypervisor.

With chain of trust, each BIOS module contains a hash of the next module in the chain. The key modules in BIOS are the Initial Boot Block (IBB), Security (SEC), Pre-EFI Initialization (PEI), Memory Reference Code (MRC), Driver Execution Environment (DXE), and Boot Device Selection (BDS). If Intel Boots Guard authenticates IBB, then IBB validates SEC+PEI before handing control to it. SEC+PEI then validates PEI+MRC which further validates the DXE+BDS modules. At this point, control is handed over to UEFI Secure Boot as explained in later sections.

Cryptographically Verified Trusted Booting

One of the most critical aspects of server security is ensuring that the boot process can be verified as secure. This process provides a trusted anchor for all subsequent operations such as booting an OS or updating firmware. The iDRAC boot process is verified using a silicon-based Root-of-Trust to meet recommendations in NIST SP 800-147B (BIOS Protection Guidelines for Servers) and NIST SP 800-155 (BIOS Integrity Measurement Guidelines).

Signed Firmware Updates

Enhanced firmware authentication is embedded within many third-party devices that provide signature validation using their own Root-of-Trust mechanisms. This prevents the possibility of a compromised third-party update tool from loading malicious firmware into devices like NICs or storage drives (and bypassing the use of signed Dell update packages). Many of the third-party PCIe and storage devices that are shipped with PowerEdge servers use a hardware Root-of-Trust to validate their respective firmware updates.

PowerEdge servers have digitally signed firmware updates for several generations to assure that only authentic firmware is running on the server platform. The firmware packages are digitally signed using SHA-512 hashing with ECDSA-384 encryption for the signature for all key server components including firmware for iDRAC, BIOS, PERC, I/O adapters and LOMs, PSUs, storage drives, FPGAs, and backplane controllers. In addition, iDRAC firmware updates are now signed with a Post-Quantum Cryptography (PQC) signature alongside the existing ECC signature. During an iDRAC firmware update, if the update package has PQC signatures, both the existing ECC and new PQC signatures are validated; the PQC signatures are validated using the Dell Security Processor framework. If either signature check fails, the update is aborted. iDRAC scans firmware updates for all other components using the existing ECC signature validation. To verify the authenticity of the firmware running on the system, the silicon-based Root-of-Trust is employed, and current signatures are compared to expected signatures. Any firmware package that fails validation is aborted and an error message is logged into the Lifecycle Controller Log (LCL) to alert IT administrators.

If any firmware in any device is suspected of malicious tampering, IT administrators can roll back many of the platform firmware images to a prior trusted version stored in iDRAC. Retain two versions of device's firmware on the server—the existing production version (N) and a prior trusted version (N-1).

Non-Root Support

iDRAC10 firmware ensures that all internal iDRAC processes only grant access according to the principle of least privilege access, a core security concept. This approach provides protection against programming flaws, intrusion attempts, and malware. The protection ensures that access is only granted to those who need it and that systems or users cannot access files or hardware that are outside the scope of their defined roles. For example, the process that provides Virtual KVM support should not be able to change fan speeds. If Virtual KVM Support user roles have different access privileges from Fan Speed Controls roles, it becomes easier for the system to prevent attacks from propagating from one process to another.

NOTE: When root user logs in using Public Key Authentication (PKA), FCP is bypassed.

When FCP is enabled, following actions are not allowed:

- Log in to iDRAC through any UI except IPMI over-LAN interface which uses CLI with default user credentials
- Log in to iDRAC through OMM app using Quick Sync-2

RSA SecurID Two-Factor Authentication (2FA) iDRAC10 Datacenter

RSA SecurID can be used as another means of authenticating a user on a system. The iDRAC10 with the Datacenter license and firmware supports RSA SecurID as another two-factor authentication method.

- [Using iDRAC10 RSA SecurID 2FA](#)This document goes through how to configure iDRAC10 to enable RSA SecurID 2FA on local users.

Active Directory

If your company uses the Microsoft Active Directory software, you can configure the software to provide access to iDRAC, allowing you to add and control iDRAC user privileges to your existing users in your directory service. This is a licensed feature.

You can configure user authentication through Active Directory to log in to the iDRAC. You can also provide role-based authorizations, which enable an administrator to configure specific privileges for each user. For full instructions on configuring Active Directory, see iDRAC user's guide.

NOTE: By default STARTLS on port 389 is configured. The connection protocol can be reconfigured to LDAPS on port 636 using Redfish or the RACADM command `racadm set idrac.ActiveDirectory.Connection LDAPS`.

The Certificate Settings page in iDRAC GUI is used to configure the digital certificate that is used during initiation of TLS/SSL connections when communicating with the Active Directory (AD) server; these communications use LDAP over TLS/SSL (LDAPS). When certificate validation is enabled, it is necessary to upload the certificate of the Certificate Authority (CA) that issued the certificate that is used by the AD server during initiation of TLS/SSL connections. The CA's certificate is used to validate the authenticity of the certificate provided by the AD server during TLS/SSL initiation. For more information, see the [Integrate iDRAC with Microsoft's Active Directory](#) whitepaper.

Table 13. Certificate Validation

Certificate Validation	
Select Enabled to validate the SSL certificate of your Active Directory server.	iDRAC always uses LDAP over Secure Socket Layer (SSL) while connecting to Active Directory. By default, iDRAC uses the CA certificate available in iDRAC to validate the SSL server certificate of the domain controllers during the SSL handshake and provides security. You can disable the certificate validation for testing purposes, or when you choose to trust the domain controllers in the security boundary without validating their SSL certificates.

LDAP

iDRAC provides a generic solution to support Lightweight Directory Access Protocol (LDAP)-based authentication. This feature does not require any schema extension on your directory services.

To make iDRAC LDAP implementation generic, the commonality between different directory services is used to group users and then map the user-group relationship. The directory service-specific action is the schema. For example, they may have different attribute names for the group, user, and the link between the user and the group. These actions can be configured in iDRAC.

NOTE: By default STARTLS on port 389 is configured. The connection protocol can be reconfigured to LDAPS using Redfish or the RACADM command `racadm set idrac.ldap.connection LDAPS`. The port can be reconfigured using redfish or the RACADM command `racadm set idrac.ldap.port 636`.