Security Features in the integrated Dell Remote Access Controller (iDRAC)

Dell iDRAC delivers key security features, comprehensive management and enhanced functionality for Dell PowerEdge servers.

Dell OpenManage Engineering Team

April 2016

Contents

This document is for informational purposes only and may contain typographical errors and technical inaccuracies. The content is provided as is, without express or implied warranties of any kind.

Introduction

The integrated Dell Remote Access Controller (iDRAC) provides comprehensive, agent-free management for Dell PowerEdge servers. Dell's philosophy for iDRAC security is "continuous improvement." All security items detailed in the iDRAC6 Security White Paper from July 2010 are part of the foundation for iDRAC7 and iDRAC8 security. In addition to many new functions and easy-to-use features, iDRAC7 and iDRAC8 continue to include the following key security features:

- A wide range of connectivity options and configurable features to further secure your iDRAC, including signed firmware updates
- Built-in Hidden Root Key (provides a number of Trusted Platform Model (TPM)-like features to iDRAC)
- Credential Vault
- Field Service Debug Authorization Facility
- Decommissioning and re-provisioning support
- Managing Web Server Certificates on iDRAC

For more updates and more information, visit delltechcenter.com/idrac

NOTE: The use of "iDRAC" in this paper denotes iDRAC7 for Dell's 12th-generation PowerEdge servers and iDRAC8 for 13th-generation servers, unless otherwise specified.

Features and Options for a more secure iDRAC

Dell offers many options to choose from to strengthen the security of iDRAC. Securing access to critical network resources is a priority. iDRAC implements a range of security features that includes:

- Custom signing certificate for Secure Socket Layer (SSL) certificate
- iDRAC provides SHA-2 2048-bit self-signed certificates
- Signed firmware updates
- User authentication through Microsoft Active Directory, generic Lightweight Directory Access Protocol (LDAP) Directory Service, or locally administered user IDs and passwords.
- Two-factor authentication using the Smart–Card logon feature. The two-factor authentication is based on the physical smart card and the smart card PIN.
- Single Sign-On and Public Key Authentication.
- Role-based authorization, to configure specific privileges for each user.
- SNMPv3 authentication for user accounts stored locally in the iDRAC.
- User ID and password configuration.
- Default login password modification.
- Set user passwords and BIOS passwords using one-way hash format for improved security.
- SMCLP and web interfaces that support 256-bit and 40-bit encryption (for countries where 128-bit is not acceptable), using the TLS 1.2 standard.
- Session time-out configuration (in seconds).
- Configurable IP ports (for SNMP, SMTP, HTTP, HTTPS, SSH, Telnet, Virtual Console, and Virtual Media).
- Secure Shell (SSH) that uses an encrypted transport layer for higher security.
- Login failure limits per IP address, with login blocking from that IP address when the limit is exceeded.
- Limited IP address range for clients connecting to iDRAC.
- Dedicated Gigabit Ethernet adapter available on rack and tower servers (additional hardware may be required).

For additional information on these items, and how to configure and implement, please refer the current iDRAC User Guide. Some features require iDRAC Enterprise licensing.

Dell Best Practice regarding iDRAC connectivity

It is Dell's recommendation that iDRACs are connected to a separate management network. They are neither designed nor intended to be connected to the public internet. Doing so could expose the connected system to security and other risks for which Dell is not responsible. Along with locating DRACs on a separate management subnet, users should isolate the management subnet/vLAN with technologies, such as firewalls, and limit access to the subnet/vLAN to authorized server administrators.

Firmware signing

In response to concerns in the IT security field and among security-conscious customers about the threat of malicious firmware, starting with iDRAC7, firmware updates are signed and verified. Updates are signed near the end of the firmware build process. The signatures are verified when firmware updates are later applied by customers. The signature generation and verification processes are as specified by the US Digital Signature Standard (FIPS-186-3). This verification results in a high level of assurance that iDRAC will only run firmware designed and delivered by Dell.

Keeping pace

Dell server technology is keeping pace with the threat landscape by following a process of continuous security improvement and innovation. Our focus is on good security features and on secure development practices, while minimizing customer impact and providing high value. Firmware code signing and verification exemplifies this approach.

The build process: signing firmware

Dell uses industry-standard cryptographic hashing and private/public key encryption technologies to sign and verify the IDRAC firmware. The iDRAC firmware contains the public (verification) key. The engineering team that designs and builds the iDRAC includes a group known as Product Group Release Engineering (PGRE). The PGRE team, while small, supports a wide variety of engineering activities. In particular, the team supports a single-purpose signing server. The signing server accepts unsigned firmware images, signs them with the private (signing) key, and returns the signed image. During this process, the private key never leaves the signing server. Because the signing server is maintained by a small group, the private key is only accessible to a handful of trusted individuals.

In addition to the actual iDRAC firmware, a number of other binary images that the iDRAC Lifecycle Controller technology manages are also signed. These other signed components are the driver pack (used during OS deployments), embedded diagnostics, and the iDRAC user interface that is accessible during boot (formerly known as the Unified Server Configurator).

Field update: verifying the firmware signature

As part of applying a field update of iDRAC, the existing firmware performs the cryptographic inverse of the signing operation, using the public key that is built into the existing firmware's image. If the signature verification fails, the update process is aborted.

Firmware downgrades

The firmware signing scheme allows firmware downgrades as long as the downgraded version is signed with the same key that the existing firmware is verifying against. In the unlikely event of a compromised private key (see following section), firmware downgrades are not allowed. At Dell, we realize that allowing firmware downgrades can be viewed as a security weakness. We have deliberately chosen to allow downgrades in order to maintain ease-of-use and supportability. If the threat landscape or customer demands require it, we may change this behavior in the future.

In the unlikely event of a compromised private key

The iDRAC firmware's signing scheme allows a new signing key to be introduced if the existing key is compromised. A new signing key is introduced by creating a version of iDRAC firmware that:

- Has the new signing key built into the image, and,
- is signed by the existing key

Because the new firmware is signed with the existing key, existing systems will determine that it is authentic; subsequently, it will only allow firmware versions to be signed with the new key.

Built-in unique Hidden Root Key

The iDRAC contains several cryptographic acceleration engines, one of which is a 128/256-bit AES engine. The other cryptographic engines are: ECC 160/256, RSA 1024/2048, DES/TDES, a random number generator, SHA1, and SHA-256. Along with these engines, each iDRAC contains a unique binary value burned into the silicon. This 256-bit unique value is hidden from software. When commanded, the iDRAC firmware will instruct the AES engine to use this number for encryption or decryption operations. The iDRAC8 takes advantage of this capability to create a "root of trust" within iDRAC's early boot code.

Hidden Root Key: keeping secrets confidential

The HRK is cryptographically composed from three sources: a 256-bit fused random value in iDRAC's CPU unit, a Public Key that is contained in iDRAC's boot block, and either the value True or the value False, depending on whether the code in the boot block was signed with the public key. (The boot block is a small amount of persistent memory that contains the initial portion of iDRAC's boot code).

Because of the way in which the HRK is calculated, a high degree of assurance is provided that the HRK for a particular iDRAC will take on a different value if code not signed by Dell is executed on that iDRAC's CPU, rather than if the code is signed by Dell.

Therefore, data encrypted by the HRK when the iDRAC is running Dell signed firmware can't be decrypted by the HRK of firmware supplied by an attacker. This means that if the flash storage chip used by iDRAC were to be accessed directly (for example, by soldering wires to it), the data within is encrypted and is therefore not accessible.

Hidden Root Key: uniquely identifying a particular server

Because the HRK is unique to every iDRAC, it provides a means to uniquely identify an iDRAC in a cryptographically robust manner.

Uniquely identifying a server can be important scenarios such as these:

- In organizations where the threat of rogue servers is possible
- As a basis for building a chain of trust for:
    - Validating that hardware components of a system are authentic
    - Confirming that the firmware and firmware configuration of the various devices on the system are unchanged and authentic

    - Clearing all aspects of customer data and customer-supplied configurations when a system is repurposed or decommissioned

- Combining servers in clusters or groups with high assurance against rogue or spoofed servers

Summary: implementing features Trusted Platform Module (TPM)-type features

The HRK allows the iDRAC to implement a number of features normally associated with the TPM, but in a way that does not add additional cost to the system, and also preserves all of the TPM's capabilities that customers might use. For more on ciphers and encryption, please see the addendum at the end of this paper for a full list of cryptographic functions supported by iDRAC7 and iDRAC8.

Credential vault

It is possible to de-solder an industry-standard flash memory device from a printed circuit board and obtain a copy of its contents. Dell iDRAC deals with this threat with its Credential Vault, which encrypts sensitive data before moving it into the iDRAC's flash memory. Private keys are a good example of the kind of sensitive data that the Credential Vault holds; this will expand to hold other items in future versions.

Protected storage

The cryptographic community uses the term "Protected Storage" to mean a storage facility that resists unauthorized attempts to read data contained within. A few well known examples of Protected Storage facilities are: smart cards, the "PStore" facility in Microsoft® Windows® 2003 and Windows XP, the BitLocker® facility in more recent Windows versions, and the TPM v1.2.

iDRAC uses the HRK mentioned previously, to encrypt sensitive information such as the private keys associated with a user-generated SSL certificate which the iDRAC then stores. Dell's iDRAC, as well as previous iDRAC generations, support generating Certificate Signing Requests (CSRs) that can be given to a certificate authority (CA). For a nominal fee a CA will generate an SSL certificate using the CSR. The resulting SSL certificate can then be uploaded to iDRAC8.)

Someone trying to steal a credential from iDRAC, by either de-soldering the flash memory chip or by installing a rogue firmware on iDRAC (assuming the attacker figures out a way to get around the firmware verification feature in iDRAC), would be thwarted if/when trying to read such credentials, because they are encrypted.

Field Service Debug authorization facility

As with many devices that incorporate embedded firmware, iDRAC8 has a number of built-in debugging features. Generally, these debug capabilities are only turned on and used during product development. On rare occasions, it may be necessary to debug iDRAC8 once it is out of the development environment. The typical alternatives of:

- Not allowing substantive debugging after the product has shipped, or
- Having a mechanism where the manufacturer can access the debug features via an undocumented mechanism

Both alternatives are problematic. iDRAC solves these problems by implementing a mechanism in which both the customer and Dell explicitly authorize debugging on a *particular* iDRAC and explicitly authorize debugging starting and ending at specific times.

Benefits

This innovation gives the customer control over their system. It protects the customer from having a back door into their system, which anyone with the right knowledge could exploit. By design, Field Service Debugging requires that each iDRAC is separately authorized. Consequently, if the token that enables debug on a particular system were stolen, other iDRAC's are not vulnerable. Further, tokens expire so a stolen or lost token only remains a threat until its expiration date. (Additionally, placing a token on an iDRAC requires the "diagnose" privilege.)

This feature preserves the ability to thoroughly debug iDRAC in the field if necessary, helping to speed problem resolution and helping Dell to continuously improve our products.

New security features in iDRAC8

Decommissioning and re-provisioning support

In a previous generation of Dell PowerEdge servers, iDRAC supported erasing all data stored since the iDRAC was deployed. This feature is typically used when a server is decommissioned or repurposed. Now, iDRAC extends this feature by also wiping the non-volatile memory in which the server BIOS stores data.

The result is that customers can sell, reuse, or throw away their servers knowing that data stored by the BIOS and by iDRAC have been erased and can't be mined by anyone who later gains possession of the server.

For a more detailed look at this feature please refer to the following white-paper: System Erase in Dell 13th-Generation PowerEdge Servers

Managing Web Server Certificates on iDRAC

Dell's integrated Dell Remote Access Controller (iDRAC) uses web server certificates (also known as SSL certificates) to establish and maintain secure communications with remote clients. Web browsers and command line utilities, such as RACADM and WS-Man, use these SSL certificates for server authentication and establishing an encrypted connection.

There are several options available to secure the network connection using web server certificates. iDRAC's web server has a SHA2 2048 bit self-signed SSL certificate by default. The self-signed certificate can be replaced with a custom signing certificate or a certificate signed by a well-known Certificate Authority (CA). Whichever method is chosen, once iDRAC is configured and the SSL certificate is installed on the management stations, SSL enabled clients can access iDRAC securely and without certificate warnings.

Note: Customers who upgrade from an older version of iDRAC to a version supporting 2048-bit self-signed certificates will need to generate a new self-signed certificate. iDRAC preserves any existing certificates during the upgrade process. To generate a new certificate, run the RACADM command "racadm sslresetcfg".

For a more detailed look at this feature please refer to the following white-paper: Managing Web Server Certificates on iDRAC

Summary

The Dell product security philosophy is to design and engineer systems for security while maintaining a balance with customer usability and value. We believe a process of continuous security improvements best serves our customers' needs. We continue to deliver on this philosophy in our current generation of PowerEdge servers, as well as looking for additional ways to enhance security features in the next generation.

Appendix A: Supported SSL Cipher Suites

For iDRAC8 we maintain the highest level of effort to provide the top level of secure communications, while still protecting interoperability and minimizing impact. Multiple levels of cipher strengths are provided from which to choose. This enables greater flexibility when the server enters negotiations with the client.

For iDRAC8, the list of supported ciphers when 128-bit (default) or higher is configured are:

DHE-RSA-AES256-SHA	TLSv1	Kx=DH	Au=RSA	Enc=AES(256)	Mac=SHA1
DHE-RSA-CAMELLIA256-SHA	TLSv1	Kx=DH	Au=RSA	Enc=Camellia(256)	Mac=SHA1
AES256-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=AES(256)	Mac=SHA1
CAMELLIA256-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=Camellia(256)	Mac=SHA1
EDH-RSA-DES-CBC3-SHA	TLSv1	Kx=DH	Au=RSA	Enc=3DES(168)	Mac=SHA1
DES-CBC3-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=3DES(168)	Mac=SHA1
DHE-RSA-AES128-SHA	TLSv1	Kx=DH	Au=RSA	Enc=AES(128)	Mac=SHA1
DHE-RSA-SEED-SHA	TLSv1	Kx=DH	Au=RSA	Enc=SEED(128)	Mac=SHA1
DHE-RSA-CAMELLIA128-SHA	TLSv1	Kx=DH	Au=RSA	Enc=Camellia(128)	Mac=SHA1
AES128-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=AES(128)	Mac=SHA1
SEED-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=SEED(128)	Mac=SHA1
CAMELLIA128-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=Camellia(128)	Mac=SHA1
IDEA-CBC-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=IDEA(128)	Mac=SHA1
RC4-SHA	TLSv1	Kx=RSA	Au=RSA	Enc=RC4(128)	Mac=SHA1
RC4-MD5	TLSv1	Kx=RSA	Au=RSA	Enc=RC4(128)	Mac=MD5