DELPHIX

Securing Delphix

December, 2017

Securing Delphix

You can find the most up-to-date technical documentation at:
docs.delphix.com The Delphix Web site also provides the latest product updates.
If you have comments about this documentation, submit your feedback to: infodev@delphix.com

Delphix Corp.
1400 Seaport Blvd, Suite 200
Redwood City, CA 94063

Securing Delphix

This section outlines standard hardening techniques that you should apply to every Delphix Engine in your environment.

The Delphix Engine delivers its functionality through a web-based graphical user interface (GUI), web-based RESTful APIs, and a command line interface (CLI) accessed over SSH. All three interfaces leverage the same accounts and privileges scheme.

Access to the Delphix Operating System (DxOS) is restricted to Delphix Support and Delphix Professional Services, and can be controlled through the Support Access Control mechanism, described below. Customer-specific software installations and modifications to the DxOS are not required or supported.

Communications to/from connected systems (“sources” and “targets”) should be limited to required ports, and encryption should be used wherever possible.

Security Principles

The Delphix approach is based on:

Embrace separation of duties: Isolate and compartmentalize capabilities and privileges and never give or concentrate access to a single role.

Apply the principle of least privilege: Users should obtain only those privileges needed to do their jobs and only for as long as they are needed.

Use an open, simple design: Make security mechanisms simple and easy to use, and rely on proven, peer-reviewed solutions.

Use a layered defense: Provide no single point of failure; if one layer fails to catch an error, catch it in another layer.

Use complete mediation and authentication: Control and check every access point every time.

Use fail-safes: Deny access when not explicitly authorized. Prevent faults from causing an opportunity to compromise.

Protect data at rest and data in motion: Utilize common security protocols as well as features of the source database and database software to protect data at all times.

Minimize the attack surface: Present the minimum sockets, services, webpages, and accounts necessary to operate.

Don’t rely on obscurity: Be secure even if everything but the key is known.

Audit and monitor everything: Provide a tamper-proof trail of evidence.

Leverage the environment: Design the Delphix Engine to leverage the security features offered by databases, operating systems, storage devices, and networks.

Anticipate external Attack Vectors: Combat attacks sourced from connected systems.

Enforce strong credentials: Define and enforce password policies.

Software Updates

Keeping Software up to date is an important part of any hardening plan. Delphix software releases are cumulative and include bug fixes, new features, and security improvements. In addition, Delphix releases hotfixes, procedures, and workarounds for critical vulnerabilities.

Patch Annually

To stay up to date, patch your Delphix Engine at least once per year. If you do not, you might have to upgrade twice to get the latest releases, and your old installed version will not be able to receive vulnerability fixes.

If possible, patch more frequently. Depending on the version you are upgrading from/to, you may be able to avoid or defer the reboot sequence, which defers downtime for your virtual databases (VDBs). This allows you to patch outside of downtime windows.

Subscribe to Delphix Notifications

Delphix issues email notifications when critical vulnerabilities are discovered. Registered support accounts will automatically receive these notifications. To ensure that you receive these notifications:

- Register at least two Delphix Admins with Delphix Support
- Add Delphix Support accounts when Delphix Admins leave the company

System Configuration

There are a number of configuration options available in the System Administration area that help you to secure the Delphix Engine. You can configure these during installation through the setup wizard, or later by accessing the Delphix Setup screens. For detailed instructions, refer to **Setting Up the Delphix Engine**.

Maintain System Time with NTP

Establish at least one, but preferably three, corporate NTP servers and sync your Delphix Engine to them. This ensures that audit and error messages display the correct time.

When configuring Delphix Engine on VMware, be sure to configure the NTP client on the ESX host to use the same servers that you enter here. On a vSphere client, the NTP client is set in the **Security Profile** section of the configuration process.

Enable Phone Home

Phone Home service will send critical information about the Delphix Engine to Delphix Support using HTTPS, on a periodic basis. Use of a Web Proxy Server is fully supported. Phone Home data allows Delphix Support to proactively detect Delphix Engines affected by critical vulnerabilities.

Register Your Delphix Engine

Registration is fast and easy, and you can do it with or without Internet connectivity from the Delphix Engine. Failing to register the Delphix Engine will impact its supportability and security in future versions.

Enable LDAP for Authentication

The LDAP protocol is used by enterprise authentication services. Enabling LDAP authentication allows your Delphix Engine to leverage the password control features of these products, such as expiration, lockout, and complexity.

Import your LDAP server certificate into your Delphix Engine, and enable SSL/TLS.

Enable SMTP and/or SNMP Monitoring

When the Delphix Engine encounters errors, it issues **alerts**. Configure SMTP and/or SNMP to forward **alerts** t o your central monitoring system.

User Management

Secure User Management is best achieved by integration with your centralized authentication service. Once integration is complete, create LDAP authenticated named users to facilitate separation of duties, least privileges, and auditing. Disable the out-of-the- box generic DELPHIX_ADMIN and SYSADMIN accounts.

Use LDAP for Authentication

As described under System Configuration above, enable LDAP authentication to leverage your enterprise authentication service and enable SSL/TLS to secure LDAP connections.

Create Named Users

Do not create generic functional accounts such as “QA,” “DEV,” or “TEST.” Such accounts will not leave a proper audit trail and violate the separation of duties principle. Instead, create LDAP authenticated named users.

For additional information, see **User Privileges for Delphix Objects**.

Assign Least Privileges

Restrict the **delphix_admin** and **sysadmin** roles to 1-2 trusted named users each. These roles are highly privileged and must be carefully managed. These roles typically map to a **DBA** and **System Administrator** res pectively.

For subordinate users who need to refresh VDBs, assign “Data Operator” privilege on the VDB and “Reader” privilege on the dSource.

For subordinate users who need to provision new VDBs from dSources, assign “Provisioner” privilege on the dSource and “Provisioner” privilege on the Group to which they will assign the VDB.

For additional information about the Delphix privilege model, see **User Privileges for Delphix Object s** and **Adding Delphix Users and Privileges**.

Consider Delphix Self-Service Functionality

The Delphix Self-Service functionality is targeted towards developer and tester self-service, and it contains a more sophisticated privilege model. With this functionality, Delphix Self-Service users do not have access to the Administrator GUI.

Administrators can define multiple data sources as a complete template. They also allocate server resources as a “data container.” The end user has the ability to update data from the source, from peers using the same source, and from prior images of the source that they have created.

Disable DELPHIX_ADMIN and SYSADMIN

Once you have established named Delphix Administrators and Systems Administrators, disable the out-of-the-box `sysadmin` and `delphix_admin` accounts. You can disable accounts through the CLI.

Delphix Operating System (DxOS)

Delphix Support accesses the DxOS for deep diagnostics and troubleshooting. Access to the Delphix Engine requires access to your network. Typically this is granted via shared troubleshooting sessions over Webex, with full transparency.

If desired, you can enable an additional control so that access can only take place when you provide a token. The **Support Access Control** feature provides this control.

If you disable Support Access, do not have the token, and you are unable to login as a system administrator, it can become impossible for Delphix Support to login to repair your system. Example: the management stack crashes, the login system becomes unavailable, and you have disabled Support Access and do not have the token. For this reason, Delphix strongly recommends leaving Support Access enabled at all times. If you wish to disable access, generate a unique token once a month and place it in a secure location separate from the Delphix Engine.

Disable Support Access (Optional)

Support Access Control is managed as a system administrator in the Server Setup area.

- When set to DISABLED, it is impossible for Delphix Support to login to the DxOS.
- When set to ENABLED (the default), Delphix Support can login to the DxOS.
- When set to ENABLED and a calendar time is set and a token generated, Delphix Support can only login with the token during that timeframe, which you provide. Generate the token once/month for an entire month and store it in a secure location separate from your Delphix Engine. When requested, provide the token through a secure means: in your support ticket, via email or SMS to a trusted entity, etc.

GUI Security

Securing the Delphix GUI is similar to securing other web consoles.

Reduce Inactive Session Timeout to 15 minutes

You can do this with a CLI command on a per-user basis by modifying the sessionTimeout Property of the User object.

For example:

```
myhost.delphix.com>  cd  user
myhost.delphix.com user> select delphix_admin
myhost.delphix.com user 'delphix_admin'> update
myhost.delphix.com user 'delphix_admin' update *> set sessionTimeout=15
myhost.delphix.com user 'delphix_admin' update *> commit
```

The default is 30 minutes.

Use a URL from your Domain and Create a Signed Certificate

Do not use IP Addresses to access your engine. Create a hostname and DNS entry, such as "**delphix1.mycompany.com**".

Delphix Professional Services can assist you in converting the engine from a self-signed certificate to a signed certificate that maps to your domain name.

Disable HTTP Access

To protect your credentials in flight and connection with the engine, disable HTTP or configure HTTP to redirect connections to HTTPS. For instructions on configuring HTTP and HTTPS, see **Changing+HTTP+and+HTTPS+Web+Connections**.

Network Security

Review the official documentation for the full list of required ports, *which depends on your database vendor*. Open only those ports that are required. The following table only lists generic requirements; you will need additional ports to integrate with databases.

> Open Only Required Ports.

General Port Allocation

The Delphix Engine makes use of the following network ports irrespective of the type of database objects on it:

General Outbound Port Allocations

Protocol	Port Numbers	Use
TCP	21	Passive FTP connections from the Delphix Engine to the Delphix FTP server. Used for sending logs to Delphix Support.
TCP	25	Connection to a local SMTP server for sending email.
TCP/UDP	53	Connections to local DNS servers
UDP	123	Connection to an NTP server
TCP/UDP	389	Standard access to an LDAP server
TCP/UDP	636	Secure access to an LDAP server
TCP	1023	Used to complete the passive FTP connection from the Delphix Engine to Delphix Support. These ports are not used if Delphix uses a proxy server to connect to the Internet.
TCP	8415	A Delphix replication source will connect to the replication target using this destination port

General Inbound Port Allocations

Protocol	Port Number	Use
TCP/UDP	22	SSH connections to the Delphix Engine
TCP	80	HTTP connections to the Delphix GUI
TCP	443	HTTPS connections to the Delphix GUI

TCP	8415	A Delphix replication target will accept incoming connections to this port from the replication

Related Links

- **Network and Connectivity Requirements for Oracle Environments**
- **Network Access Requirements for SQL Server**
- **Network and Connectivity Requirements for SAP ASE Environments**
- **Network and Connectivity Requirements for DB2 Environments**

Source Database Security

Choose Minimum Privileges for Delphix DB User

The Delphix Engine requires a DB user (**delphix_db)** on your source databases. This account is necessary to detect the state of the source and stay in sync. Do not give unnecessary privileges to this user. Leverage the script provided by Delphix in the hostchecker bundle to create a user with the minimum required privileges. The DB user (e.g., **delphix_db**, which is the example used on this page) account can have the same or different user name on each of your source databases.

Protect the Delphix DB User Password

Because the **delphix_db** user has access to sensitive data dictionary information, take steps to protect access to this account.

Use Database Encryption Functionality to Encrypt Sensitive Data at Rest

Database vendors provide tools to encrypt data at rest. This encrypts data on disk in order to provide protection at rest. Use the database encryption on your source databases to encrypt sensitive or all data. Delphix integrates seamlessly with database encryption to provide protection at rest.

Choose Encrypt and Compress when Linking

Delphix provides encryption capabilities when linking against your source databases. Encrypting while linking can lead to higher CPU utilization and higher throughput. This overhead can be reduced by selecting both the Compress and Encrypt options, in order to compress the data before it is encrypted. Train your Delphix Administrators to choose both options.

Source and Target Host Security

Oracle on UNIX

Delphix support for Oracle on UNIX requires an OS account (**delphix_os**) on source database servers and on target servers that will host virtual databases or files. The Delphix Engine uses SSH to send commands to this user, which performs operations on the host. Some of these commands require elevated privileges.

> There is no actual requirement that the account be named **delphix_os** on both sources and targets. You can name the account anything you want; you can also use separate accounts on every source and target.

- **Restrict su - delphix_os to named Delphix Admins and System Administrators**

  Other users of the system do not need access to the delphix_os user. Your Delphix Admins and System Administrators should retain su ability to facilitate troubleshooting.

- **Use SSH Key exchange to allow the Delphix Engine to communicate with targets**

  Implement public/private key exchange instead of username/password. This allows you to keep the password of **delphix_os** completely secret.

- **Put delphix_os on password rotation**

  Rotate the delphix_os password in accordance with your enterprise security policy for application software accounts. You should either:

  - implement SSH Key exchange prior to placing delphix_os on password rotation, or
  - script CLI commands to update the password inside the engine as part of the rotation process.

  Delphix Professional Services can assist you in integrating Delphix with your enterprise password rotation system.

- **Restrict elevated privilege commands to the lowest level needed**

  The Delphix Engine uses elevated privileges to provide core features as well as optional features. The Delphix docs describe in detail which privileges are absolutely necessary, as well as techniques for further restricting the commands that can be used.

  The Delphix Engine ships with support for “sudo” as the privilege elevation system, but also allows for integration with third-party and custom centralized privilege managements systems.

Windows

Delphix support for SQL Server requires two OS accounts for Windows:

- **delphix_src** – used on the source database server
- **delphix_trgt** – used on the servers which host Virtual Databases Both are required for the

Validated Sync target

> There is no actual requirement that the account be named **delphix_src/delphix_trgt**. You can name the account anything you want; you can also use separate accounts on every source and target. Finally, you can create a single account for use everywhere, but this is not recommended since it violates separation of duties.

Restrict privileged commands to the lowest level needed

The Delphix user or domain account should have exactly the privileges required in the Delphix documentation. Do not grant additional privileges.

Put delphix_src and delphix_trgt on password rotation

Change the user or domain account password at regular intervals or in accordance with security policies for application software accounts. Use CLI scripts to quickly modify the password across the Delphix ecosystem. Delphix Professional Services can assist you in scripting and integrating Delphix with your enterprise password rotation system.

Use minimum privileges on your SMB share

Consult **http://technet.microsoft.com/en-us/library/cc754178.aspx** to understand how shared folder privileges work. Use the minimum privileges.

Use Windows Authentication for SQL Server

SQL Server allows authentication via Windows or Mixed mode. Mixed mode allows authentication via Windows or SQL Server.

Windows authentication is more secure; it uses Kerberos security protocol, provides password policy enforcement with regard to complexity validation for strong passwords, provides support for account lockout, and supports password expiration. **http://msdn.microsoft.com/en-us/library/ms144284.aspx**

Replication Security

Choose Encrypt and Compress when Replicating

Delphix provides encryption capabilities when replicating data from one Delphix Engine to another.

The Delphix Engine intelligently compresses data before encryption when both **Encrypt** and **Compress** options are selected. This leads to lower CPU utilization and higher throughput compared to using encryption alone, with the same level of protection. See **Configuring Replication** for details.

Train your Delphix Administrators to choose both options.

Masking Sensitive Data

Encryption does not protect data that is accessed through applications and database clients, the most likely attack vector. Masking sensitive data before it gets to non-production systems is a critical tool in the security arsenal.

Delphix provides an add-on masking product for simple cost-effective integration. It is also possible to integrate Delphix with any masking technology. See **Delphix Masking** for details.

There are many topologies to consider, and the complete explanation of their pros and cons is outside the scope of this document. Delphix Professional Services can assist you in analyzing various masking solutions.

Audit Logs

Review Audit Logs Monthly

Conduct a monthly review of audit logs on your Delphix Engine. Pay particular attention to provisioning operations of unmasked databases, which creates new copies of your production data. See **Accessing Audit Logs** for instructions.

Forward Audit Logs to Central Server via SYSLOG

Forward audit logs to a central audit server using SYSLOG techniques. Delphix Professional Services can assist you with scripts that facilitate this. See also **Setting SysLog Preferences** for configuration instructions.

Additional Topics

The Delphix Engine provides robust, enterprise-quality security controls. Performing the steps listed in this document will allow you to easily bring your Delphix Engines into compliance with your organization's security policies.

Perform a Yearly Audit

At least once annually, audit one or more Delphix Engines to ensure compliance with your security policies.

Port Scan

Delphix fully supports network security scans, using a tool of your choosing.

Security Testing

Many companies require security testing of applications in their environment using a Port Scanner or other Security Penetration Test tools. Delphix supports the use of these security tools with the application credentials available for the engine (e.g., delphix_admin). The Delphix Engine is a close appliance, and OS credentials on the appliance are not provided for these tests.

Security Banner

Configure your custom security banner, which will be displayed to all users prior to login. For example: "You are accessing a secure system."

Virtual Database Security

The Delphix Engine provides advanced storage capabilities and automation to allow rapid provisioning of virtual databases (VDBs), which use only a fraction of the physical storage used by full database copies. Nonetheless, a VDB is equivalent to a physical database **and must be properly secured like any other database**.

By far the most dangerous attack vectors in the Delphix ecosystem are the same ones that existed pre-Delphix: unauthorized access to your non-production systems containing sensitive production data. **You must perform all the same actions to harden virtual databases as you would to harden physical clones**.

For information on securing your virtual databases, consult vendor-specific material and security guides.