Quest Authentication Services 4.0.3
Slide Index

• Learning Objectives- Slide #3
• Product Functional Overview- Slides#4,5
• Management Console for Unix-Slide#12-14
• Troubleshooting Checklist-Slide 18
• QAS Debugging Methods-Slides#19-24
• Common Support Scenario- Slides#28-38
• Prerequisites for Contacting Support-Slide#39
Learning Objectives – How To Diagnose & Troubleshoot

- Upon completion of this lesson, the student should have an understanding of the product use case, the product components, and be able to troubleshoot issues. The student should also be able to collect all the necessary information that support would require to diagnose an issue in the event a service request needs to be raised.
Quest Authentication Services (QAS) enables organizations to extend the security and compliance of Active Directory with many other Enterprise platforms such as Unix, Linux, and Mac. It addresses the compliance need for cross-platform access control, the operational need for centralized authentication and single sign-on, and enables the unification of identities and directories for simplified identity and access management.

QAS securely and conveniently eliminates the need for manual per-system identity administration, User and Group NIS maps, and password synchronization scripting.

The product components that allow for such functionality include:

- QAS Daemon
- Group Policy (VGP)
- QAS Control Center
- Management Console for Unix (MCU)
QAS Daemon

• The QAS Daemon (vasd) is the core Authentication Services client daemon. The vasd daemon must be running on Unix clients in order for QAS to operate correctly. When started, vasd utilizes Kerberos to authenticate against Windows domain controllers using credentials that are established at the time when the computer was joined to the domain. Vasd then uses Kerberos encrypted LDAP sessions to query and cache Unix user and group information that is necessary for the scalable and secure operation of the nss vas and pam vas modules.

• The vasd daemon provides several important features:

  • **Secure Access Control:**

  • Do to the way in which PAM and NSS subsystems operate, most LDAP-based Unix account management solutions require that anonymous or public access be granted to the Unix account properties. Quest Authentication Services does not require a reduction in access control policy for Unix account properties. Since vasd authenticates a Unix host much like an Active Directory domain computer, there is no need to provide anonymous access to any information used by the QAS client.

  • **Secure LDAP without SSL:**

  • Once the Unix host is joined to the domain, vasd is able to authenticate as a domain computer and encrypt LDAP communications with domain controllers using a Kerberos session key. The Quest Authentication Services client never generates any plain-text LDAP traffic, and does not require the administrative overhead of SSL certificate distribution to Unix clients or running Active Directory with SSL enabled.
QAS Daemon

- **Scalability:**
  - Because of the way that PAM and NSS subsystems operate, most LDAP based Unix account management solutions generate excessive numbers of LDAP connections and LDAP search requests. This can result in dramatically increased network traffic and load on the Windows Active Directory domain controllers. `vasd` establishes a single connection that is used to proxy all information requests for processes that call the NSS interfaces. At the same time, `vasd` is able to perform intelligent caching of frequently used information so that LDAP traffic is reduced to the absolute minimum.

- **Disconnected Operation:**
  - `vasd` maintains a persistent cache of frequently queried information. This allows the system to continue to operate in environments where the network connection to the Active Directory server is unreliable or completely unavailable. Disconnected operation is particularly useful for remote users that are frequently not connected to the network.
Group Policy

Vintela Group Policy (VGP) allows administrators to use Microsoft Group Policies to manage the configuration and settings of non-Windows based operating systems and applications. VGP allows Group Policies to become a single integrated tool for managing resource configuration in your enterprise, Windows and non-Windows alike.

Microsoft Group Policy provides excellent policy-based configuration management tools for Windows, but until VGP, Group Policy did not provide the same capability to manage Unix resources. In a mixed-platform environment, Unix administrators had to use manual configuration, scripting, or a completely separate tool for configuration management of non-Windows operating systems and applications. VGP allows administrators to consolidate configuration management tasks and reuse the Group Policy functionality of Microsoft Windows Server to manage Unix operating systems and Unix application settings.

The following is a list of core design objectives for VGP that allows the preservation of both Microsoft Group Policy and Unix configuration management concepts:

- Re-use the Windows server-side components and extensibility to provide a tightly integrated and familiar deployment architecture and administrative interface.
- Provide a flexible client-side group policy implementation on Unix that matches the extensible functionality of the corresponding Windows client-side group policy implementation.
- Provide Windows server-side and Unix client-side extensions that surface functionality that is flexible enough to be useful in solving most common Unix configuration management problems.
QAS Control Center

Quest Authentication Services consists of plug ins, extensions, security modules and utilities spread across many different operating systems. The primary function is managing the global configuration and licensing for the QAS Daemon. The QAS Control Center pulls those parts together and provides a single place for you to find the information and resources you need.
QAS Control Center

What's New in Management Console for Unix

- **Quest Privilege Manager for Sudo Integration**
  - Support for centralized access policy management, remote sudo plugin installation and configuration, keystroke logging and replay, and reporting.
  - Learn more

- **Role-based Access to the Management Console**
  - Active Directory users and groups can now be granted access to the management console and given limited use of console features by means of roles.
  - Learn more

- **New Command Line Utilities**
  - Quest One Management Console for Unix now provides Unix command line utilities and Windows PowerShell cmdlets that enable you to script common local Unix user and group management tasks. For example:
    - Create a user account across multiple systems
    - Change a local Unix user’s password across multiple systems
    - Create a script to collect data for custom reports
    - Learn more

- **Other New Management Console Features**
  - Automatic Profiling of Hosts - keep the management console up-to-date with changes made on your hosts outside the console
  - Use of sudo for tasks requiring elevated privileges on the host, and limit tasks in the console
  - Improved management of AD users and groups, including group membership
    - Learn more

- **Upgraded from Identity Manager for Unix 1.0?**
  - If you have upgraded from Quest Identity Manager for Unix 1.0 to Quest Management Console for Unix 2.0, be aware of the following:
    - Passwords cached by the supervisor account or AD users with console access were not migrated during the upgrade process due to changes in encryption. Users will have to re-enter their passwords for hosts they manage the next time they perform tasks on the hosts, and choose to re-enter their credentials again on the server.
    - If you plan to use the management console with Quest Privilege Manager for Sudo, it’s important to re-profile all hosts, including any managed policy server hosts where Quest Privilege Manager is installed.
    - Existing Active Directory users and groups granted access to the management console are added to the Manage Hosts role, giving them access to the features they had before the upgrade.
QAS Control Center
Management Console for Unix

The Management Console for Unix (MCU) allows you to centrally manage Quest Authentication Services agents running on Unix, Linux and Mac OS X operating systems.

The management console has the following capabilities:

- Remotely deploy QAS agent software to systems.
- Manage local user and group accounts.
- Configure and control account mappings from local users to Active Directory accounts.
- Report on a variety of security and host access related information.

The management console is compatible with any operating system. Once installed, its web interface can be accessed by any browser. The default port utilized is 9443, however is configurable.
Management Console for Unix
Management Console for Unix

<table>
<thead>
<tr>
<th>User</th>
<th>UID</th>
<th>Comment (DECS)</th>
<th>Host</th>
<th>AD User</th>
</tr>
</thead>
<tbody>
<tr>
<td>root</td>
<td>25</td>
<td>Dacl login daemon</td>
<td>slico10-h.example.com</td>
<td>EXAMPLE\textbackslash{}smith</td>
</tr>
<tr>
<td>bill</td>
<td>2151</td>
<td></td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>bin</td>
<td>1</td>
<td></td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>daemon</td>
<td>2</td>
<td>Daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>ftp</td>
<td>40</td>
<td>FTP account</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>games</td>
<td>12</td>
<td>Games account</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>gdm</td>
<td>50</td>
<td>Gnome Display Manager daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>halddaemon</td>
<td>101</td>
<td>User for hald daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>jason</td>
<td>2152</td>
<td>Jason Smith</td>
<td>slico10-h.example.com</td>
<td>EXAMPLE\textbackslash{}jason</td>
</tr>
<tr>
<td>jason_local</td>
<td>2150</td>
<td>local unix account for jason f</td>
<td>slico10-h.example.com</td>
<td>EXAMPLE\textbackslash{}jasonf</td>
</tr>
<tr>
<td>lo</td>
<td>4</td>
<td>Printing daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>mail</td>
<td>8</td>
<td>Mailer daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>man</td>
<td>13</td>
<td>Manual pages viewer</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>messages</td>
<td>100</td>
<td>User for D-BUS</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>news</td>
<td>9</td>
<td>News system</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>nobody</td>
<td>65534</td>
<td>nobody</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>rts</td>
<td>74</td>
<td>NTP daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>postfix</td>
<td>51</td>
<td>Postfix Daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>root</td>
<td>0</td>
<td>root</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
<tr>
<td>sshd</td>
<td>21</td>
<td>SSH daemon</td>
<td>slico10-h.example.com</td>
<td></td>
</tr>
</tbody>
</table>
## Quest Authentication Services Known Issues

<table>
<thead>
<tr>
<th>Solution</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOL82767</td>
<td>SOL82767 QAS 4.0.3 Maintenence Fix ChangeLog and download link</td>
</tr>
<tr>
<td>SOL69850</td>
<td>SOL69850 Upgrading 3.5.2.x to 4.0.x causes Unix-enabled accounts to become invalid.</td>
</tr>
<tr>
<td>SOL82822</td>
<td>SOL82822 Joins using keytab failing: &quot;ERROR: Failed to establish host credentials: VAS_ERR_CRED_NEEDED: Unable to find a keytab entry&quot;</td>
</tr>
<tr>
<td>SOL87880</td>
<td>SOL87880 Mobile Mac Users Slowness Reported</td>
</tr>
<tr>
<td>SOL90913</td>
<td>SOL90913 Mapped users stop working after upgrade to Quest Authentication Services 4.0.3.91</td>
</tr>
<tr>
<td>SOL90898</td>
<td>SOL90898 OAT not removing local users and groups</td>
</tr>
<tr>
<td>SOL84182</td>
<td>SOL84182 Autosys job cannot get user ID</td>
</tr>
<tr>
<td>SOL86035</td>
<td>SOL86035 Group-override not working after upgrade to 4.0.3</td>
</tr>
<tr>
<td>SOL88788</td>
<td>SOL88788 /etc/security/user not correctly setting the system stanza when other repositories are defined</td>
</tr>
<tr>
<td>SOL88233</td>
<td>SOL88233 vastool flush command hangs or is extremely slow</td>
</tr>
<tr>
<td>SOL82127</td>
<td>SOL82127 In 4.0.3 vastool list users output is different</td>
</tr>
<tr>
<td>SOL84826</td>
<td>SOL84826 error: ld.so.1: .vpgtool: fatal: libstdc.so.5: open failed: No such file or dir</td>
</tr>
<tr>
<td>SOL87020</td>
<td>SOL87020 vas_status.sh error &quot;grep: illegal option -- e&quot;</td>
</tr>
<tr>
<td>SOL90874</td>
<td>SOL90874 AD accounts not locking after bad attempts limit is reached</td>
</tr>
</tbody>
</table>
Install Locations

QAS Daemon

Configuration Files:
/etc/opt/quest/

Binary Files:
/opt/quest/

Data and run files:
/var/opt/quest/

QAS Control Center

C:\Program Files\Quest Software\Authentication Services

Management Console for UNIX

Windows - C:\Program Files\Quest Software\Management Console for Unix
Unix - /opt/quest/mcu
Other locations

Where are log files?

QAS Daemon
vasd uses the Unix standard syslog for outputting logging messages. Most often information can be found in the /var/log/messages file.

QAS Control Center
The control center stores the logs to the current logged in user’s home folder. For example:

%SystemDrive%:\Users\username\AppData\Local\Quest Software\Authentication Services\Logs

Management Console for Unix

- On Windows XP/2003 Server:
  %SystemDrive%:\Documents and Settings\All Users\Application Data\Quest Software\Management Console for Unix\resources

- On Windows 2008 Server/Vista/7:
  %SystemDrive%:\ProgramData\Quest Software\Management Console for Unix\resources

- On Unix/Mac:
  /var/opt/quest/mcu/resources
Troubleshooting Checklists

The Quest Support Portal provides several valuable knowledgebase articles to address common product issues:

SOL87741 - How to enable debug on Quest One Management Console for Unix (MCU)?

SOL66941 - Unable to login

SOL85924 - vastool manpage for QAS 4.0.3

SOL66941 - Unable to login

SOL89397 - Tivoli Desktop not authenticating

SOL88101 - Can not login due to access group policy not being applied

SOL70429 - When a Domain Controller (DC) goes offline or is unreachable does Quest Authentication Services (QAS) use another DC?

SOL90535 - How do you provide the UNIX password hash to legacy systems using vasyp?
QAS Debugging Methods

The debugging of Quest Authentication is broken down into 4 key areas.

**vasd:** when experiencing performance issues, stability issues or issues with any of the following sections

For issues that require debugging it is always best practice to have debug set to level 5. At this level LDAP query information including messages sent/received and their size are recorded.

- This setting is located within `/etc/opt/quest/vas/vas.conf`
- It can be controlled by the following syntax:

  ```bash
  # vastool configure vas vasd debug-level <value, 1-5, or blank to remove>
  ``

For example:

  ```bash
  # vastool configure vas vasd debug-level 5
  ```
QAS Debugging Methods

**pam_vas**: For issues surrounding authentication problems, the debugging of the pam_vas area should be analyzed.

Add "debug trace" to the end of the appropriate lines (pam_vas modules only), in the appropriate file(s). This example line:

```
login auth sufficient /opt/quest/lib/security/$ISA/pam_vas3.so get_tgt create_homedir get_nonvas_pass
```

then becomes

```
login auth sufficient /opt/quest/lib/security/$ISA/pam_vas3.so get_tgt create_homedir get_nonvas_pass debug trace
```

On HP and Solaris, the file is `/etc/pam.conf`. On Linux, Redhat it is `/etc/pam.d/system-auth`, and Suse would require modifying the specific service file in `/etc/pam.d/`. It is only necessary to add these lines to the specific service that the authentications are failing with. The above example was for the login service. If the service is not known or does not have a specific entry, place the ‘debug trace’ on all appropriate lines.
QAS Debugging Methods

**nss_vas**: for issues with user and group name resolution

Use `nss_vas` to fulfill system `getpwnam/getpwent/getpwuid/getgrgid/getgrnam/getgrent` calls.

Enable debug in the `nss_vas` section of `/etc/opt/vas/vas.conf`, and is set as follows:

```
# vastool configure vas nss_vas enable-debug [true|false]
```

The output from `nss_vas` is recorded to the file `/tmp/nss_vas.log`.
QAS Debugging Methods

**aix_vas:** for both authentication and user/group resolution on AIX systems

AIX uses LAM, so the pam logging equivalent is set as follows:

```bash
# vastool configure vas aix_vas auth-debug [true|false]
```

and the nss section is handled as follows:

```bash
# vastool configure vas aix_vas nss-debug [true|false]
```

Capturing VAS Debug Messages through Syslog

All information goes to syslog. Configure Syslog as follows:
1. Make sure syslog is running (`ps -ef | grep syslogd`). If not, start it instead of the `kill -HUP` in Step 4.
2. Verify that `/etc/syslog.conf` contains the following line:
   ```
   *.debug <tab> <tab> /tmp/vas_debug.log <or other location>
   ```
3. Create the empty log file:
   ```
   # touch /tmp/vas_debug.log <or other location>
   ```
4. Configure QAS with debug level 5:
   ```
   # vastool configure vas vasd debug-level 5
   ```
5. HUP the syslog process number `kill -HUP <syslogd pid>`, this will cause it to restart.
6. The log file should now contain the debug logs from vasd.
QAS Debugging Methods

**vastool**: used when output directly to screen is needed, for quick analysis of the results. If something isn’t spotted here, then vasd and/or PAM/NSS would then be needed.

Also, you can run vastool and output debug information to the screen, by running it as follows:

```
# vastool -d <level of debug> <Rest of commands>
```

Be aware when running vastool with debugging, if it calls vasd for any work, like during a flush, vasd will not output its debug information.
QAS Debugging items to look for

Key words to look for in debug files:

- `<username>` customer is working with
- Failed will bring you to an error(s) usually in communication with AD that were denied etc...
- authentication will bring you to the authentication log(s) for a user authentication
- timeout usually found around communication attempts to AD. Will expose an down server etc... busy system etc...
- error will bring you to error(s) in the QAS daemon
Snapshot scripts

QAS Snapshot

The “snapshot” script is now included with all versions of Quest Authentication Services. It contains many pieces of information which support utilizes when troubleshooting a case. Items such as sanity checks, DB validations, various network/OS test and Quest specific configurations are all included within the package.

To run this script, simply execute the following script as root:

```
# /opt/quest/libexec/vas/scripts/vas_snapshot.sh
```

This will create an archive file in the /tmp folder in the following format:

```
vas_snapshot.<hostname>.YYYY-MM-DD_HH-MM-SS.tar.gz
```

When a service request is required, this file should always be attached to the service request so that Support can analyze the contents. Having this information readily available will help to reduce the resolution time.
Snapshot scripts

QAS Status

We include a status script will all installation. The snapshot automatically run this as well, but is always recommended to run this as it will give you invaluable information about the system’s state and could save you hours of troubleshooting. The script is executed when you run as root:

```
# vastool status
```

And the actual script can be found here:

```
/var/opt/quest/libexec/vas/scripts/vas_status.sh
```
Enabling Logging MCU

To enable the debug log
1. Stop the Quest One Management Console for Unix service
2. Open the custom.cfg file for editing. The custom.cfg file is in the application data directory:
   - On Windows XP/2003 Server:
     \%SystemDrive%:\Documents and Settings\All Users\Application Data\Quest Software\Management Console for Unix\resources
   - On Windows 2008 Server/Vista/7:
     \%SystemDrive%:\ProgramData\Quest Software\Management Console for Unix\resources
   - On Unix/Mac:
     /var/opt/quest/mcu/resources
3. Add these system variables to the custom.cfg file:
   - -Dlog4j.configuration=log4j-debug.xml
   - -Djcsi.kerberos.debug=true
4. Save the custom.cfg file.
5. Start the Quest One Management Console for Unix service.

By default, the debug logs are saved in the application data directory at:
   - On Windows XP/Windows Server 2003:
     \%SystemDrive%:\Documents And Settings\All Users\Application Data\Quest Software\Management Console for Unix\logs
   - On Windows Vista/Windows 7:
     \%SystemDrive%:\ProgramData\Quest Software\Management Console for Unix\logs
   - On Unix/Mac platforms:
     /var/opt/quest/mcu
Installation Errors

Symptom
Some ftp programs do not recognize certain package extensions, thus transferring the files in ASCII format will corrupt the binary files rendering them unusable.

Problem description
Errors on file sizes or other errors pointing to corrupted install files.

Solution
Put the ftp program being used in binary file transfer mode when moving the install files to the final destination.
Common Support Scenarios

General Joining Errors

To troubleshoot joining issues, re-run the join commands this time capturing debug information by adding the following:

-d5 right after vastool and then the rest of the command, for example: /opt/quest/bin/vastool -d5 –u administrator join example.com 2>&1 | tee join.debug.log

When completed review the log file for errors. If a support case is required, compress the file and attach to the support case, along with any relevant environment details and the QAS Snapshot.

Specific Symptom
vastool cannot locate the proper domain controllers.

Problem description
Cannot join, or KDC not found, or there are firewalls in the environment, or limited DNS information.

Solution
If there are no DNS entries for the domain, or firewall rules are prohibiting nix servers from receiving the information it requires, examine the firewall rules that exist.

Common Support Scenarios

**Example setup (for testing and reproducing):**

Modify the /etc/hosts entry adding 192.168.0.45 dc01.example.com dc01 example.com

This will circumvent DNS for the connection, ensuring both the domain and the domain controller resolve to the proper IP address.

Use iptables to block network calls:

- `iptables -A output -j accept -p tcp --dport 88 -d dc01.example.com`
- `iptables -A output -j accept -p tcp --dport 389 -d dc01.example.com`
- `iptables -A output -j accept -p tcp --dport 464 -d dc01.example.com`
- `iptables -A OUTPUT -j DROP`

This simulates a DMZ, opening only the essentials ports.

The following command configures QAS to use only TCP for communications, disabling UDP.

```
# vastool configure vas libvas use-tcp-only true
```

Re-run the join command with the following syntax:

```
vastool -u administrator join example.com dc01.example.com
```

This join will utilize a specific domain controller, using TCP only. This solution should enable a successful join, allowing domain users to authenticate on the UNIX server.
Common Support Scenarios

User Authentication Errors

This section addresses issues for systems where QAS has been successfully joined, but users still cannot authenticate.

Note: When you run vastool with “-u host/” as listed below, the command needs to be performed as root or through the use of sudo.

Specifying host/ as the user indicates to QAS to use the /etc/opt/quest/vas/host.keytab which is readable only by root for security reasons as it contains the keys (password hashes) to the corresponding Active Directory computer object.

For most task this is the quickest and easiest way to retrieve data from Active Directory.
Common Support Scenarios

All User Authentications Failing

Verify the computer host object in Active Directory and the host.keytab which allows access to that object. If nothing is returned, it was successful. Run as root:

```
# /opt/quest/bin/vastool -u host/ kinit -S host/
```

If it fails, you need to recreate the computer host object and host.keytab file. Run as root:

```
# /opt/quest/bin/vastool -u administrator create host/
```

If that fails, proceed to create debug output by adding -d5 as discussed before:

```
# /opt/quest/bin/vastool -d5 -u administrator create host/ 2>&1 | tee vastool.create.debug.log
```

Once completed, send the file created and an explanation of the issue to Support along with the QAS snapshot. If it succeeded, try the kinit command again. If it continues to fail add debug to the command and then send the file to support.

```
# /opt/quest/bin/vastool -d5 -u host/ kinit -S host/ 2>&1 | tee vastool.kinit.host.debug.log
```

If you have installed the SFU schema extension, there might be an issue with the Global Catalog (GC). By default, SFU only extends the UID attribute to the GC, and no others.
Common Support Scenarios

All User Authentications Failing continued

When the machine is joined, and you execute as root:

# vastool -u host/ schema detect

QAS will do an extensive search for the UNIX attributes, and detect that not everything is in the GC. When one of vasd’s internal timers expires it checks GC status, it does a smaller search which will think UNIX attributes are being extended to the GC. But then when it fails to find the other attributes, vasd will remove the user from the cache.

The symptoms will be users unable to log in, and after, a “vastool list users” won’t list the tried user.
Common Support Scenarios

Single User Authentication Fails

To verify that host can log in as root or through sudo as explained in the All User Authentication Fails section. These debugging techniques assume no specific error messages is given when the login fails. It appears as though the password was wrong, or a non-existent user name was used.

On AIX, an error about terminal ownership usually means the user’s primary GID did not resolve.

The best errors usually come from Telnet or su. When using su, you need to go through the initial root -> su user first, then do su user again. When root does an su, there is no authentication, and important logon steps are skipped that might expose the problem. If root can’t su, that will be important to tell Quest Support if the following does not expose the issue.

First step:
If the user having the issue is available (can enter their password for the following command), get a new ticket from Active Directory similar to the login process, complete the following steps:
Run this as root or with sudo: /opt/quest/bin/vastool -u <username> kinit -S host/

If that takes more than a few seconds, it’s possible DNS isn’t resolving fast enough. See Troubleshooting DNS for more information.

Second step:
Run /opt/quest/bin/vastool list user <username>

If that shows the user as you expect, see Single User Listed in QAS Cache. If the user is not listed, see Users Not Listed in QAS Cache.
Common Support Scenarios

Single User Listed in VAS Cache

To determine if the user is showing up through the NSS subsystem, run:

```
/opt/quest/bin/vastool nss getpwnam <username>
```

- The values returned should display VAS for the password field. If it isn’t VAS, then you still have a local user account.
- If that does return, check the following:
  - Is shell is valid: (run the shell and see if you get another prompt).
  - Is there a UID conflict: ( `vastool user checkconflict <username>` ) See What do I do about a UID conflict?
  - Is the user allowed access: ( `vastool user checkaccess <username>` ). If this fails, you can run it with `-d5` after `vastool` to see in what section it failed.
  - Does the primary group resolve?: ( `vastool nss getgrgid <users gid entry>` ).

On AIX, a username greater than 8 characters, would cause issues. This can be seen when trying to `su` to the user, the name returned as invalid isn’t the name you just tried to `su` to, See Quest knowledgebase article SQL47719 Error on AIX: "Cannot set process credentials".

If that doesn’t return, then something is preventing VAS from answering NSS calls.
Common Support Scenarios

Single User Listed in VAS Cache

To determine if the user is showing up through the NSS subsystem, run: /opt/quest/bin/vastool nss getpwnam <username>

- The values returned should display VAS for the password field. If it isn't VAS, then you still have a local user account.
- If that does return, check the following:
  - Is shell is valid: (run the shell and see if you get another prompt).
  - Is there a UID conflict: (vastool user checkconflict <username>) See What do I do about a UID conflict?
  - Is the user allowed access: (vastool user checkaccess <username>). If this fails, you can run it with -d5 after vastool to see in what section it failed.
  - Does the primary group resolve?: (vastool nss getgrgid <users gid entry>).

On AIX, a username greater than 8 characters, would cause issues. This can be seen when trying to su to the user, the name returned as invalid isn't the name you just tried to su to, See Quest knowledgebase article SOL47719 Error on AIX: "Cannot set process credentials".

If that doesn't return, then something is preventing VAS from answering NSS calls.
Common Support Scenarios

Users Not Listed in VAS Cache

To troubleshoot users who are not listed in the VAS Cache, run:

```
# vastool -u host/ list -l user <username>
```

This tells vastool to do a user list, but pull the information directly from Active Directory, instead of the local cache. If the user does show up, it is possible vasd hasn’t pulled it into the cache yet.

Tell vastool to do a forced update of the user information, going to Active directory and putting what it finds there in the cache:

```
# vastool -u host/ list -f user <username>
```

If that doesn’t work, check for a full /var/ partition, otherwise complete the debug steps below and send the file to support along with the QAS Snapshot.

List a user from Active Directory that exists, but vastool doesn’t think is QAS-enabled. If the user doesn’t show up, run:

```
# vastool -u host/ list -la user <username>
```
Common Support Scenarios

If that shows the user, check the same attributes vastool looks at when determining whether the user should be considered VAS-enabled, run the following:

```
vastool -u host/ attrs <username> uidnumber gidnumber gecos unixhomedirectory loginshell
```

(Or the MS-SFU names if your schema was extended with Microsoft SFU)
If all those attributes show, yet a list -f user doesn’t bring the user into the cache, first, make sure the /var/ partition isn’t full, then run:

```
vastool -u host/ -d5 list -f user <username> 2>&1 | tee list.f.user.debug.log
```

Once completed, send the file to Support along with the QAS Snapshot.

If the -la doesn’t show the user, run:

```
vastool -u host/ -d5 list -la user <username> 2>&1 | tee list.ldap.all.user.debug.log
```

Send that file to Support along with the QAS Snapshot and the following attributes of the user (Get from Active Directory):

distinguishedname
cn
samaccountname
userprincipalname

Another possible issue is the computer object in AD doesn’t have read privileges to the user’s object. This must be resolved before QAS can load the user into the cache.
Prerequisites for Contacting Support
When opening a service request submit the following:

- Problem Description
- Diagnostic logs, screenshots, etc.
- Environmental details (system versions, physical/virtual hardware, federation, High Availability, architecture, etc.)
- Issue severity and business impact, timeframes, etc.
- If a performance issue, provide specific details as specified in the Notes section of this slide.
This document Authentication Services 4.0.3 Support Training contains confidential information of Dell and embodies trade secret and proprietary intellectual property of Dell. It is legally protected and shall not be copied, modified, reverse engineered, published, disclosed, disseminated outside of Dell or otherwise used, in whole or in part, without Dell’s written consent, provided, however, that you have the right to use the Document solely for your internal use and solely as necessary for you to enjoy the benefit of Services under the applicable SOW (or other agreement) you have entered into with Dell. Copyright 2012 by Dell Inc. The copyright notice does not imply publication of this document or its contents.

DELL, the E (Stylized in a sphere) logo, Dell Compellent, OpenManage, EqualLogic, PowerEdge, PowerVault and other Dell trademarks are the trademarks or registered trademarks of Dell Inc. in the U.S. and certain other countries.