JetPatch

JetPatch Log Files

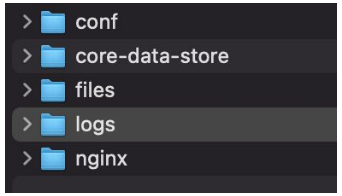
Understanding JetPatch’s Log Files

Objectives:

- What logs files exists
- Where can they be found
- How to use the Logs for basic troubleshooting
- How to set the server’s logging level

JetPatch's Manager Logs (Part 1)

- JetPatch's Manager (server) log files are located in the archive that you can download from the JetPatch console (within the logs folder)



conf
core-data-store
files
logs
nginx

- Each log has up to 10 files (by default - this is configurable), overwritten, Ordered from 1 to 9, such as the most recent is named without the number (e.g. vmanage.log)

- **Discovery Source Logs**: discovery.log.*
  These logs are helpful in understanding automatic discovery thru discover sources (e.g. WSUS, AWS, vCenter, Azure, Active Directory)

- **Patching Logs**: patching.log.*
  These are helpful in understanding problems related to remediation plans / patching

JetPatch’s Manager Logs (Part 2)

- **vmanage.log.***
  These are the logs used for aplicative logic done by the Manager server

- **worker.log.* -** very useful for connector issues

- **Apache Tomcat (Application Server) Logs**:
  Can be useful in cases of application server failures. In cases of network problems or Java problems:
  access_tomcat.<date>.log.*, catalina.<date>.log.*, localhost.<date>.log.*

- **Within the files folder (manifest and servermeta file):** includes JetPatch build**,** tomcat version, linux version, java version, memory free/full, disk free and more

Other Folders (Files and Conf)

Under the files folder, you can find a manifest file (name: MANIFEST.*number*.MF). If you open this file (with a text editor) you can get additional information on the Manager system, including version and build information (fields called 'Implementation-Version' and 'Implementation-Build'):

- Manifest-Version: 1.0
- Implementation-Title: vmanage-server
- Implementation-Version: 4.1.0
- Archiver-Version: Plexus Archiver
- Built-By: root
- Implementation-Build: 56
- Implementation-Vendor-Id: com.intigua
- Created-By: Apache Maven 3.2.1
- Build-Jdk: 1.8.0_91

Under the conf folder, you can find intigua.properties and intigua.config.xml

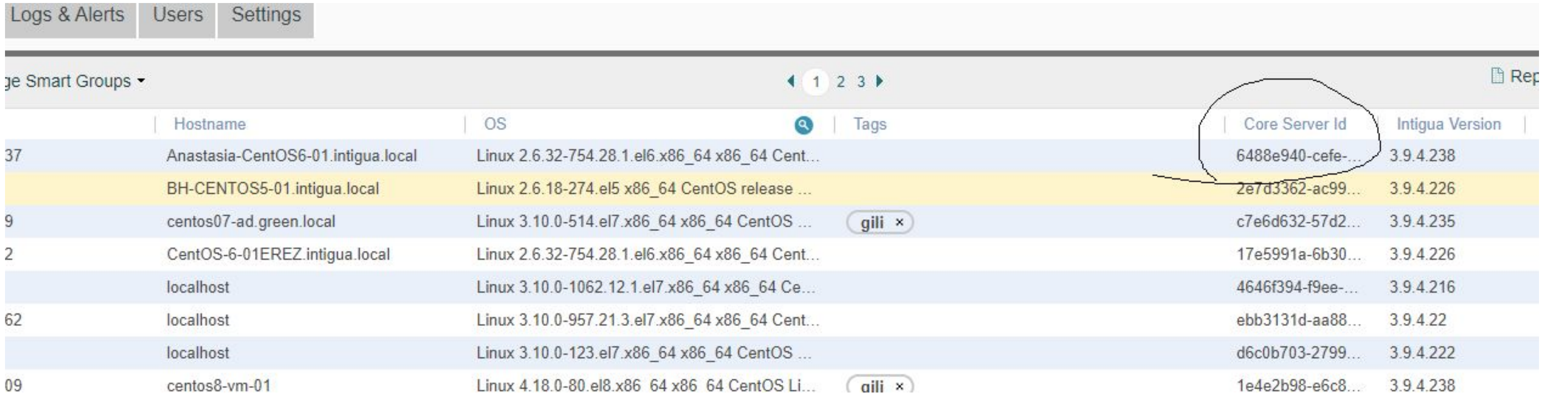
JetPatch’s Manager Logs Tips (Part 1)

<u>How to search vmanage.log.* logs files:</u>

- First find the relevant host name or IP then look for its matching core server id.
  Then you can search the Manager’s log file using this string (for example using the grep command)
  S=<core_server_id from endpoint_server table>

e.g. 2019-12-17 11:33:24,852 - ERROR [T= Worker-10][**S=4af03548**>][OP=INSTALL_VLINK][R=123257-R]

- How to find core server id for a given endpoint: Simply add the field called: ‘Core Server Id’ to the Servers tab:



	Hostname	OS	Tags	Core Server Id	Intigua Version
37	Anastasia-CentOS6-01.intigua.local	Linux 2.6.32-754.28.1.el6.x86_64 x86_64 Cent...		6488e940-cefe-...	3.9.4.238
	BH-CENTOS5-01.intigua.local	Linux 2.6.18-274.el5 x86_64 CentOS release ...		2e7d3362-ac99...	3.9.4.226
9	centos07-ad.green.local	Linux 3.10.0-514.el7.x86_64 x86_64 CentOS ...	gili ×	c7e6d632-57d2...	3.9.4.235
2	CentOS-6-01EREZ.intigua.local	Linux 2.6.32-754.28.1.el6.x86_64 x86_64 Cent...		17e5991a-6b30...	3.9.4.226
	localhost	Linux 3.10.0-1062.12.1.el7.x86_64 x86_64 Ce...		4646f394-f9ee-...	3.9.4.216
62	localhost	Linux 3.10.0-957.21.3.el7.x86_64 x86_64 Cent...		ebb3131d-aa88...	3.9.4.22
	localhost	Linux 3.10.0-123.el7.x86_64 x86_64 CentOS ...		d6c0b703-2799...	3.9.4.222
09	centos8-vm-01	Linux 4.18.0-80.el8.x86_64 x86_64 CentOS Li...	gili ×	1e4e2b98-e6c8...	3.9.4.238

JetPatch’s Manager Logs Tips (Part 2)

- Another way is to search by (grep..) the time of failure. For example: “2019-12-17 11:33”

- Focus on Error & Failure severities.
  The logs is composed of Error, Failure, Warn and if configured Debug severities
  Looking for Error and Failure log entries can help poin-point the problem.

- Once you got the relevant logs lines, you might be able to understand the cause of the failure or help JetPatch’s support team zoom into the problem

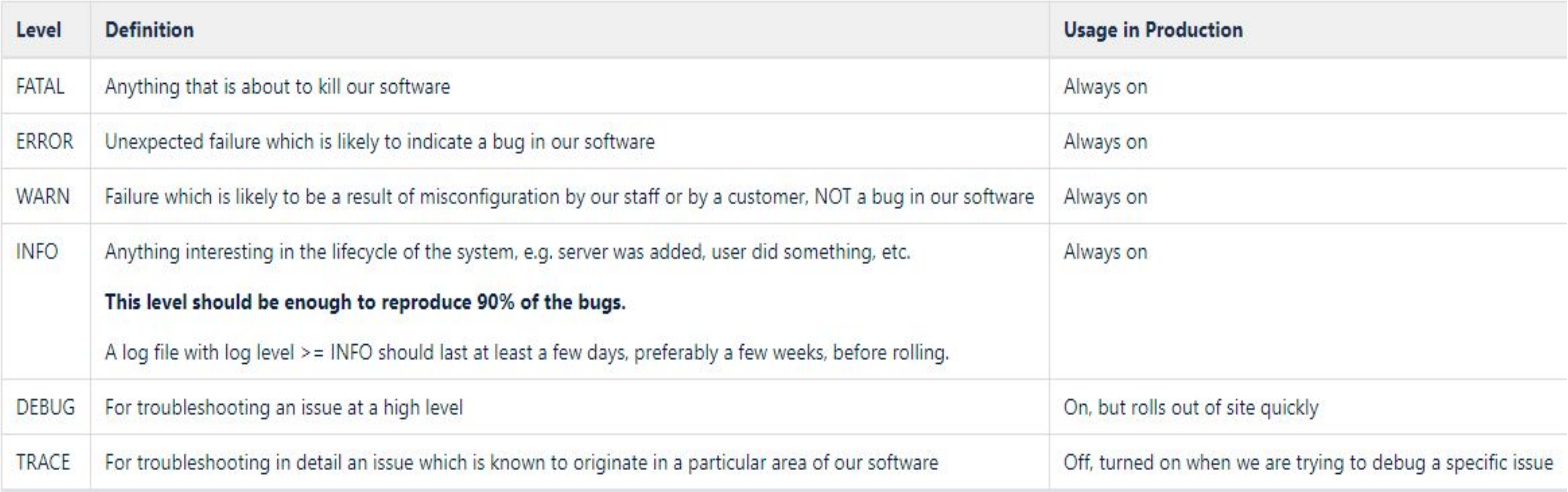
Discovery Log Example

Discovery Source permission uses, causing discovery problem, taken from discovery log:

```
2020-04-03 09:06:33,005 - ERROR [T=velopment)][AmazonClientHelper:117]- Failed to retrieve VM instances from Amazon
for region {RegionName: ca-central-1,Endpoint: ec2.ca-central-1.amazonaws.com}
com.amazonaws.services.ec2.model.AmazonEC2Exception: You are not authorized to perform this operation. (Service:
AmazonEC2; Status Code: 403; Error Code: UnauthorizedOperation; Request ID: 48dec85c-f946-4021-88c1-046f20a69e7e)
```

Severity: Error
Message: The error message can imply there is a permission problem to retrieve VM information from AWS discovery source on particular region

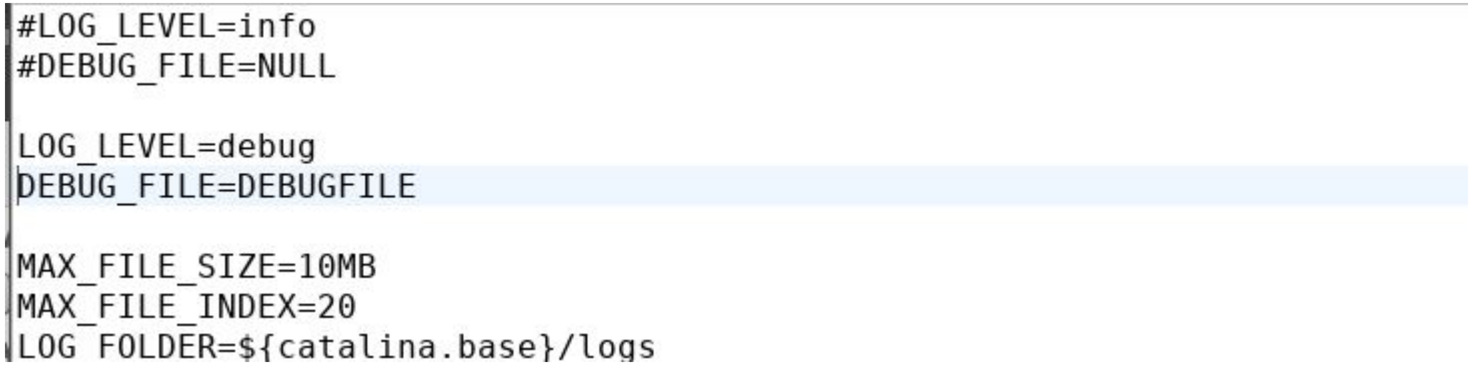
Logging Levels



Level	Definition	Usage in Production
FATAL	Anything that is about to kill our software	Always on
ERROR	Unexpected failure which is likely to indicate a bug in our software	Always on
WARN	Failure which is likely to be a result of misconfiguration by our staff or by a customer, NOT a bug in our software	Always on
INFO	Anything interesting in the lifecycle of the system, e.g. server was added, user did something, etc. **This level should be enough to reproduce 90% of the bugs.** A log file with log level >= INFO should last at least a few days, preferably a few weeks, before rolling.	Always on
DEBUG	For troubleshooting an issue at a high level	On, but rolls out of site quickly
TRACE	For troubleshooting in detail an issue which is known to originate in a particular area of our software	Off, turned on when we are trying to debug a specific issue

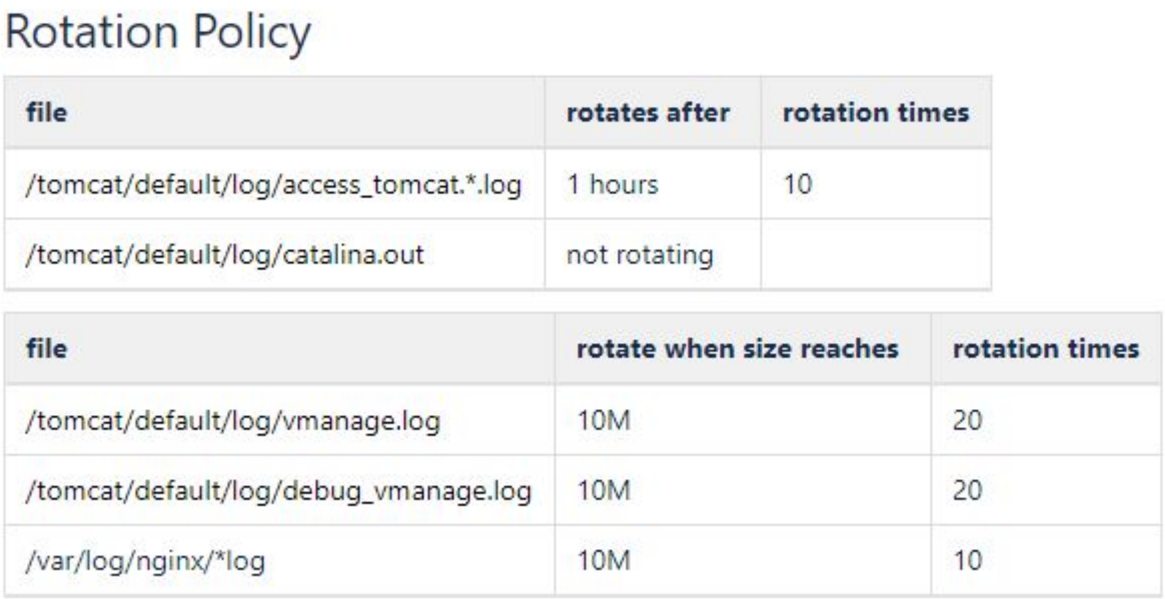
Setting the Manager’s Logging Level

- Changing the Manager’s logging level can be done by setting the logging level in the file called: usr/shr/intigua/conf/log4j.properties
- Attribute called: LOG_LEVEL
- After setting the logging level, the Manager’s application (Tomcat) needs to be restarted
- For example setting the log level to debug:



```
#LOG_LEVEL=info
#DEBUG_FILE=NULL

LOG_LEVEL=debug
DEBUG_FILE=DEBUGFILE

MAX_FILE_SIZE=10MB
MAX_FILE_INDEX=20
LOG FOLDER=${catalina.base}/logs
```

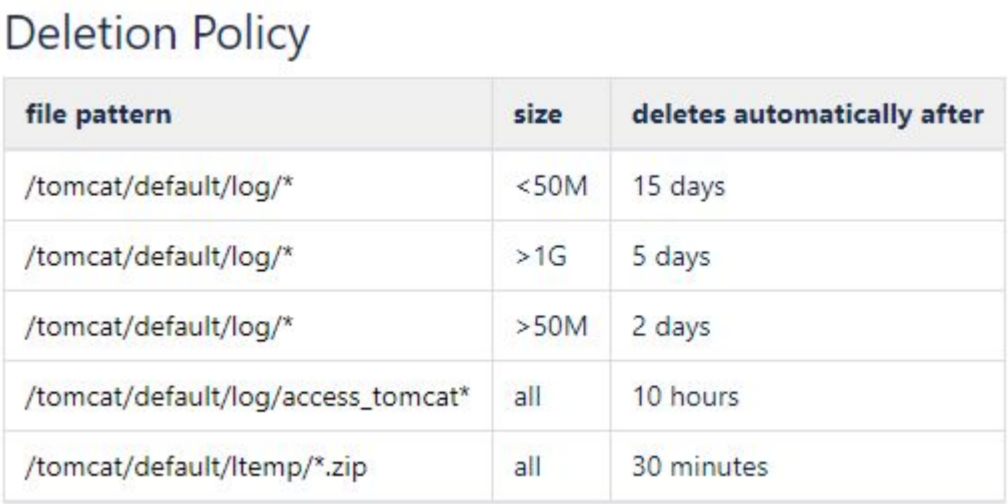
Backup



Rotation Policy

file	rotates after	rotation times
/tomcat/default/log/access_tomcat.*.log	1 hours	10
/tomcat/default/log/catalina.out	not rotating	

file	rotate when size reaches	rotation times
/tomcat/default/log/vmanage.log	10M	20
/tomcat/default/log/debug_vmanage.log	10M	20
/var/log/nginx/*log	10M	10



Deletion Policy

file pattern	size	deletes automatically after
/tomcat/default/log/*	<50M	15 days
/tomcat/default/log/*	>1G	5 days
/tomcat/default/log/*	>50M	2 days
/tomcat/default/log/access_tomcat*	all	10 hours
/tomcat/default/ltemp/*.zip	all	30 minutes

Connector Logs Tips

Connector logs can be downloaded from the console and are located on every endpoint machine under:

- **Linux**: /usr/local/intigua/vAgentManager/log
- **Windows**: \Program Files\Intigua\vAgentManager\log

Runtime Logs

- Windows
    - **vlinkprocess*.txt** - logs the connector main process operations (info/error/debug...)
    - **vlinkservice*.txt** - logs the connector service operations
- **Jetpatch_Enduser_App_log*.txt** - logs the enduser notification operations
    - Linux/Unix
- **connector*.txt** - logs the connector main process operations and connector service operations with a separate file

General Log Files

- **current_connector_log.txt** - contains the last connector runtime logs file content
- **current_Status_file.txt** - contains the last status file content with the updated data that will be sent to the Jetpatch server
- **commands_dir_file_list.txt** - contains the status of the command directory in the queue that the connector should execute
- **/components/csclient.cfg** - stored the relevant data regarding the connector communication with the Jetpatch server

Connector Logs Tips - Cont’

Specific log file in the same location called: **post_install_log***, can be used for troubleshooting connector installation issues. It captures the installation steps including various connector parameter settings:

```
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__CORE_ID__]:
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__CORE_SERVER_URL__]:
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__EXE_CMD__]:/usr/local/intigua/vAgentManager/PackageManager/vlink/vlink/bin/connector64 process
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__EXE_PATH__]:/usr/local/intigua/vAgentManager/PackageManager/vlink/vlink/bin/connector64
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__PID_FILE__]:/var/run/intiguaConnector
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__RUNLEVEL_PRIORITY__]:70
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__SERVICE_NAME__]:intigua
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__STOPLEVEL_PRIORITY__]:30
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__VERSION__]:3.9.4.238
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__VLINK_BINDIR__]:/usr/local/intigua/vAgentManager/PackageManager/vlink/vlink/bin
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__VLINK_LOG_DIR__]:/usr/local/intigua/vAgentManager/PackageManager/vlink/log
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__VLINK_PROC_NAME__]:connector64
1625  1625 2020-04-22 9:51:54.228 INFO |GENERAL:    [__VLINK_VER_PATH__]:/usr/local/intigua/vAgentManager/PackageManager/vlink/vlink_3.9.4.238
1625  1625 2020-04-22 9:51:54.229 INFO |GENERAL: Expanding vlink vars in:
"/usr/local/intigua/vAgentManager/PackageManager/vlink/vlink_3.9.4.238/bin/init_functions.sh"
1625  1625 2020-04-22 9:51:54.230 INFO |GENERAL: Expanding vlink vars in:
"/usr/local/intigua/vAgentManager/PackageManager/vlink/vlink_3.9.4.238/bin/uninstallHelper.sh"
1625  1625 2020-04-22 9:51:54.231 INFO |GENERAL: Done deploying shell scripts
1625  1625 2020-04-22 9:51:54.496 INFO |GENERAL: Enabling intigua service as systemd based service
```

Connector Log Example 1

Connector connectivity problem to JetPatch’s Manager, taken from the connector’s log:

```
local/intigua/vAgentManager/temp/Runnables/Runnables/e5f5240a-c578-4572-92ae-0f71bf4ffb7e iid=InternalDummy
timeout=3600000 policy=InternalDummy extraEnv size=8
 4815  5082 2020-04-22 13:44:09.636 ERROR|HTTP   : Http Request Failed. request (RegisterMsg: statusData = ?) error
(Failed to connect to 10.231.143.48 port 443: Connection timed out) ErrorCode:(7)
log/post_install_log.1587577312544.txt_3.9.4.226_jdk00006440000000000000000000003415513650100753017706 0ustar
rootroot[ Install Script ] Running command: /var/tmp/
```

Severity: Error
Message: Can be easily understood… HTTPS connection times out… to Manager’s IP.

Connector Log Example 2

Connector Installation problem, taken from the Manager's log:

```
2020-04-23 07:59:38,844 - ERROR [T= Worker-13][S=152a67d4>][OP=INSTALL_VLINK][R=4213-R]
[LogServiceImpl:280]- ERROR admin 1.1.1.1 Failed to install Intigua 3.9.4.235 on server 1.1.1.1
java.net.UnknownHostException: 1.1.1.1 INSTALL VLINK
```

Severity: Error
Message: The error message can imply on some sort of host resolution problem.. Can be a DNS problem or connectivity problem

Connector Log Example 3

Opposite example… of a successful connector patch installation

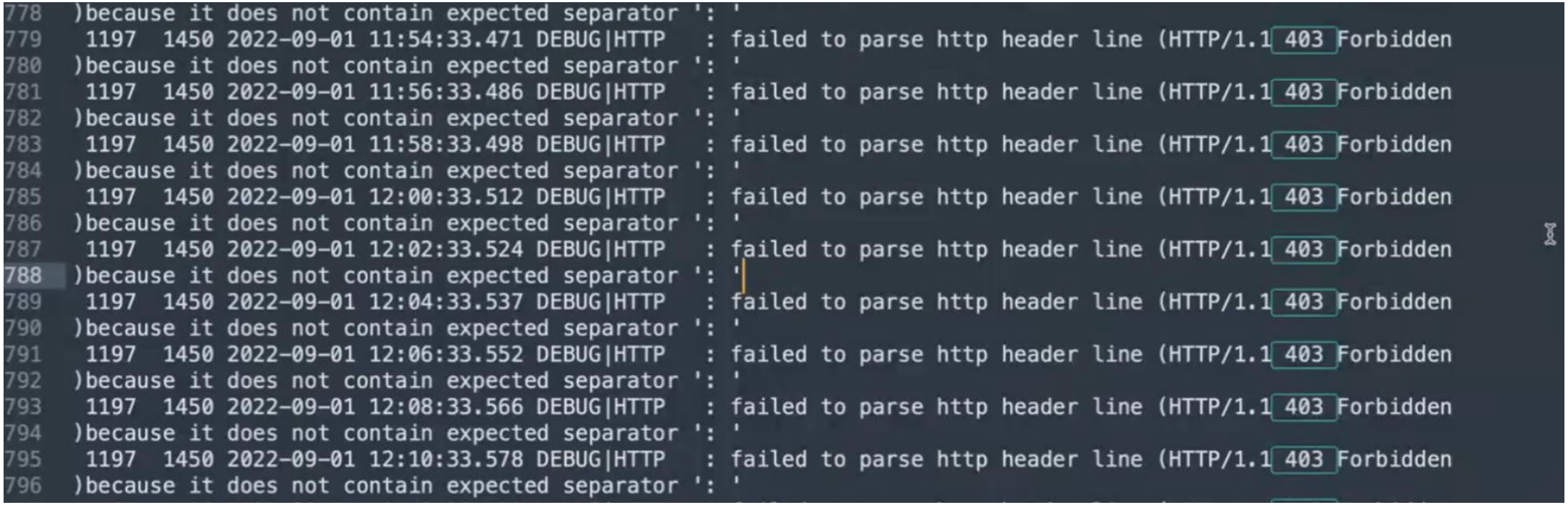
```
7636 2021-08-09 14:26:09.777 INFO |RUNNABL: Running
`C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe` args=`-File "C:/Program
Files/Intigua/vAgentManager/temp/Runnables/Runnables/335d516c-e8b5-40cc-957a-96fa125c960e/execute_patch_installation_for_single_computer.ps1" ` workDir=`C:/Program
Files/Intigua/vAgentManager/temp/Runnables/Runnables/335d516c-e8b5-40cc-957a-96fa125c960e
iid=InternalDummy timeout=10800000 policy=InternalDummy extraEnv size=0

-08-09 15:26:46.078 DEBUG|RUNNABL: Runnable task '335d516c-e8b5-40cc-957a-96fa125c960e'
SUCCEEDED. exit code=0
```

Severity: Debug

Message: Successful execution of task ‘execute_patch_installation_for_single_computer.ps1’

Connector Log Example 4

In connector logs: vAgentManager/log/connector64_[X].txt, you may see the following 403 lines



```
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 11:54:33.471 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 11:56:33.486 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 11:58:33.498 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 12:00:33.512 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 12:02:33.524 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 12:04:33.537 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 12:06:33.552 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 12:08:33.566 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
 1197  1450 2022-09-01 12:10:33.578 DEBUG|HTTP    : failed to parse http header line (HTTP/1.1 403 Forbidden
)because it does not contain expected separator ': '
```

Indicates an environmental issue. Check with network team, leveraging the relevant timestamps.

Other Log Files

Other logs: Depending on the possible sources of the problem, can be useful:

- WSUS related problems: WSUS log found on WSUS server
  Can be useful for findings MSFT WSUS related problems

- Connector related problems: Connector logs located under **`/usr/local/intigua/vAgentManager/log`** on every endpoint machine

- Database Logs (typically it is /var/lib/pgsql/<version number>/data/). If the service is running you can use the following command to get the exact path:
  - ps -ef | grep postgres

- Nginx logs located under /var/log/nginx
  These log are useful to understand issues related to https connectivity problem between the Manager and Endpoints or client (UI) machines.