DDN Collect Tool User Guide

Preface

DDN Collect is a new support utility that makes diagnostic data collection simpler and faster on SFA storage platforms. A single Python line command (or menu option) directs the tool to aggregate system status information, logs, and diagnostics into a single file transfer bundle for upload to DDN Support.

Audience

All operators of DDN SFA products.

About this Guide

This guide describes the DDN Collect Tool that is used to collect and send SFA system diagnostic data to DDN support. The latest version of this software and guide is available for download at:

- https://www.ddn.com/download/ddn-collect/

The software is also available directly through wget:

```
wget -O ddn_collect.zip https://www.ddn.com/?wpdmdl=53633
```

Related Documentation

The following documents are sources of information for this product:

- SFA OS User Guide
- SFA OS API Reference Guide

The latest version of the documentation is available on the DDN Support Portal at:

https://community.ddn.com/login

1. Introduction

The DDN Collect Tool is an easy new way to collect and send SFA diagnostics to support. It is easy to install & run and is supported on most platforms. Compatible with all existing versions of SFA OS, it allows for a simple and straightforward way to gather the necessary files without error. Please contact DDN support regarding questions about usage of this tool.

2. Glossary

The following terms are used throughout this document.

API

Application Programming Interface. The DDN SFA OS API client is included in this tool. The API client allows for programmatic access to SFA system objects and logs.

DIAGNOSTICS

DIAGNOSTICS. Sometimes referenced as DIAGS. DIAGS are a collection of files containing developer logs, kernel logs, and system crash core files. Developer logs contain state of SFA OS firmware including memory contents, including a very high resolution log file called logdisk for SFA developer analysis. Kernel logs contain syslog, dmesg, log files from /var/log, and other files. During unexpected firmware events (e.g. null pointer access) core files saved in /tmp/corefiles for each RP (for 14KX, 2 core files) as janus_core-n.m “n” is the RP number, starting with 0; “m” is the corefile number, 1 for most recent, 2 for 2nd most recent crashes. Only last 4 instances kept.

CLUI

Command-Line User Interface. The CLUI is the interface used when you ssh into a controller.

SFA Logging

References to SFA Logging are regarding the tool creating entries in the SFA system logs stating that data collection has occurred. An entry is made at the start of collection time, and another entry upon successful completion.

3. Quick Start

Follow these simple steps to quickly make use of the tool. If any previous version is installed, please delete it before installing the new version.

1. Prepare: `(delete any previous installation)`
2. Extract: `tar -zxvf ddncollect-1.7.4.tar.gz`
3. Install: `cd ddncollect-1.7.4; ./install.sh`
4. Configure: `./ddncollect configure`

   (or edit ddntools.conf manually)
5. Execute: `./ddncollect collect`
6. Upload: `./ddncollect upload`

   (only necessary if auto_upload not set)

4. Download

Please download the latest version from the following location. Improvements are continually being made to the tool and it is important to have the most up-to-date version.

Link to download location: https://www.ddn.com/download/ddn-collect/

5. System Requirements

The API client package is distributed alongside the collector tool. The SFA API server on the SFA controllers listens on port 5989 (5988 if user has enabled unsecure access via the SFAOS CLUI). The system on which this tool resides must have communication access to the SFA controllers through this port.

Additionally, the API client requires the following packages to be installed on the target system:

- Python 3.4 or higher, including development headers Python 3.6 or higher are recommended.
  - Python 2.7 is officially deprecated but still supported. Support will be dropped in a future release.It is strongly recommended that API users transition to Python 3.
- pip version 9.0.0 or higher.

- setuptools version 24.2.0
- (Linux only) OpenSSL 1.0.1 or higher is recommended. OpenSSL versions prior to 1.0.0 are not supported.
- RedHat/CentOS Package List:
    - python-devel python-pip python-setuptools libssl-devel libffi-devel python3-ptyprocess python3-pexpect python3-pkg-resources python3-six python3-yaml python3-requests
- Debian/Ubuntu Package List:
    - python-dev python-pip python-setuptools libssl-dev libffi-dev python3-ptyprocess python3-pexpect python3-pkg-resources python3-six python3-yaml python3-requests

6. Installation & Upgrade

The tool is designed to support the various systems that exist in the field. It is currently supported on RedHat/CentOS and Debian/Ubuntu. Either Python version 2 or 3 may be used; Python 3 is recommended as Python 2 is no longer supported. If Python version 2 is used, it must be at least 2.7.

PLEASE NOTE: For best performance, the tool must be installed on a system that has appropriate network connectivity to the SFA controllers. If the connection has high latency, the tool could take extremely long times to collect diagnostics.

Installation:
To install the tool simply extract the compressed tar file and execute the installation script, install.sh.

Upgrade:
If there is a previous version of the DDN Collect Tool installed, the recommended procedure is to delete the previous installation before installing the new one. There are many changes between these two versions making them incompatible with each other. The configuration file (ddntools.conf) has changes that will make the tool not work with the previous version's config file.

Notable recent changes:

- Uploads now via https to datadirect.files.com in place of ftp to DDNTSR
- Executable name change from *ddncollect.py* to simply *ddncollect*
- New command-line options allow for config file overrides and easier execution
- New SFA API client version: 12.2. Supports all SFAOS versions 3.x and above
- Now supports collecting from multiple SFA systems from a single command execution

7. Execution

Overview

Please note the changes from the previous release:

- The main executable has been renamed to ddncollect
- Configuration files from the previous version are not compatible with the new version
- Uploads now go via HTTPS to datadirect.files.com in place of FTP to DDNTSR

  IMPORTANT! In ddntools.conf, the controller configuration section has been renamed from [Controller] to [SFA0]. If multiple SFA systems need to be configured another section, labeled [SFA1] must be created, having the same variables as the initial section. If even more SFA systems need to be configured, simply follow the same pattern (e.g. add a [SFA2] section with c0_ip, c1_ip, and user_name).

```
./ddncollect help

USAGE:
The DDN Collector Tool has been created to allow admins to easily extract the system information, logs, and
diagnostics from any system to be utilized by DDN Support for quick analysis. To use the tool, first either
fill out ddntools.conf directly or execute './ddncollect configure' for prompted entry. After configuration
of the settings, execute the data collection via: 'ddncollect collect'. Finally, upload the bundle through
'./ddncollect upload'. A user-friendly menu-based option is also available that guides the user through the
process. To use this option execute the command without options or through 'ddncollect menu'.

Please contact DDN support regarding usage of this tool.

    ddncollect [command] [command-options]

commands:

    help                 Display this help
    menu                 Run the tool using a menu-based system
    install              Install the tool
    configure            Configure ddntools.conf
    settings             View settings in ddntools.conf
    test      [options]  Test password-less connections to controllers
    collect   [options]  Collect data according to the settings
    select               Select a bundle for upload
    upload               Send selected bundle to DDN
    restart   [options]  Restart the SFAOS API server

Collect Options:
  --admin_name ADMIN_NAME
                        Admin or contact name.
  --admin_phone ADMIN_PHONE
                        Admin or contact phone number.
  --admin_email ADMIN_EMAIL
                        Admin or contact email.
  --company_name COMPANY_NAME
                        Company name.
  --case_number CASE_NUMBER
                        Support case ticket number.
  --collect_clui [True|False]
                        Collect Show Subsystem Summary (and other) command
                        output.
  --collect_diags [True|False]
                        Collect SFA system engineering diagnostics.
  --suppress_sfa_logging [True|False]
                        Suppress making an entry in the SFA system log
                        regarding collection attempt.
  --verbose [True|False]
                        Verbose output.
  --directory DIRECTORY
                        Directory to store collected data.
  --auto_upload [True|False]
                        Automatically upload upon completion of collection.
  --conf_file CONF_FILE
                        Use non-default configuration file.
  --systems SYSTEMS     Comma-separated list of SFA systems to collect from as
                        configured in ddntools.conf.Name must match config
```

```
                        file section name exactly. EXAMPLE: --systems
                        SFA0,SFA1
  --core_files {1,2,3,all,lite,report}
                        Select the number of crash cores to collect. Only
                        applies when collecting DIAGS.
  --use_xz [True|False]
                        Use xz compression if available. XZ compression is only
                        available on SFAOS > 3.1.2.0 and Python 3.
  --ssh_key KEY_FILE    Use KEY_FILE as ssh key.
  --proxy PROXY         Use POXY as proxy server for bundle uploads.
  --cachepasswords      Cache passwords at start. NOTE: passwords cannot be
                        cached on Diagnostics collection. Please use ssh_key
                        option to remove password prompts for diags.

EXAMPLE: ddncollect collect --collect_clui True --suppress_sfa_logging False
--cachepasswords --systems sfa0,sfa1
```

Execution Examples

Command-line execution

The tool may be executed through strictly command-line usage, or through the menu-based system. To utilize the tool through the command line, use the 'collect' command and then add options:

```
./ddncollect collect --admin_name "My Name" --admin_phone 512-555-1212 --company_name "My Company" --case_number 123456 --collect_clui --directory /var/tmp/ddncollect --systems sfa0
```

Configure

This example configures ddntools.conf through the prompting mechanism enabled through the 'configure' command option

```
# ./ddncollect configure
Configuring options for ddncollect tool. Please answer prompts. Options will be stored in ddntools.conf

    Admin Contact Name: : Joe Admin
    Admin Phone: : 512-555-1212
    Admin Email: : myemail@mycompany.com
    Company Name: : My Company
    Case Number: : 123456
    Collect CLUI Output:  [True]:
    Collect Diagnostics:  [False]: False
    Directory to Store Data:  [/var/tmp/ddncollect]:
    Suppress Writing to SFA Controller Logs:  [True]:
    Verbose Output?:  [False]:
    Auto Upload?:  [True]:
    SFA 0 Controller 0 IP:  [10.10.10.1]: 10.36.13.80
    SFA 0 Controller 1 IP (Enter 'None' if not desired):  [10.10.10.2]: 10.36.13.81
    SFA 0 Username:  [user]:
Add additional SFA? Y/[N] :y
    SFA 1 Controller 0 IP: : 10.36.13.82
    SFA 1 Controller 1 IP (Enter 'None' if not desired): : 10.36.13.83
    SFA 1 Controller Username: : user
Add additional SFA? Y/[N] :n
Finished collecting configuration. Next recommended action: ./ddncollect collect
```

Collect

This example collects CLUI, and API data and automatically uploads the data to DDN.

```
# ./ddncollect collect --cachepasswords
DDN Collect application invoked
Action chosen: collect
Option overrides: (These options take precedence over values in configuration file.)
    cachepasswords=True
```

```
Data collection requested.

Passwords not cached. To speed things up, passwords will be entered now.
Enter passwords to the controllers. Passwords will be cached in memory only and not written to disk.

Password for SFA0 user@10.25.80.16:
Password for SFA1 user@10.25.80.32:
Passwords entered successfully.
Gathering API data from SFA0.
[API data written to data/2021-09-21-14-30-DDN-Test-123456-SFA0-sysinfo.tar.gz]
[Elapsed time: 00:00:40]
Gathering API data from SFA1.
[API data written to data/2021-09-21-14-30-DDN-Test-123456-SFA1-sysinfo.tar.gz]
[Elapsed time: 00:00:31]
Gathering Show Sub Sum output from SFA0 controller 10.25.80.16.
[Elapsed time: 00:00:11]
Gathering Show Sub Sum output from SFA0 controller 10.25.80.17.
[Elapsed time: 00:00:12]
Gathering Show Sub Sum output from SFA1 controller 10.25.80.32.
[Elapsed time: 00:00:05]
Gathering Show Sub Sum output from SFA1 controller 10.25.80.33.
[Elapsed time: 00:00:05]
Gathering bundles into single file...
[Adding DDN Collect Tool execution logs to bundle]
[Adding data/2021-09-21-14-30-DDN-Test-123456-SFA0-sysinfo.tar.gz]
[Adding data/2021-09-21-14-30-DDN-Test-123456-SFA1-sysinfo.tar.gz]
[Adding data/2021-09-21-14-30-DDN-Test-10.25.80.16-123456-SFA0-text-pre_service.txt]
[Adding data/2021-09-21-14-30-DDN-Test-10.25.80.17-123456-SFA0-text-pre_service.txt]
[Adding data/2021-09-21-14-30-DDN-Test-10.25.80.32-123456-SFA1-text-pre_service.txt]
[Adding data/2021-09-21-14-30-DDN-Test-10.25.80.33-123456-SFA1-text-pre_service.txt]
Support files written to 2021-09-21-14-30-DDN-Test-123456.tar.gz.
Auto Upload option is enabled. Bundle will now be uploaded.
[Sending data to DDN using HTTPS to datdirect.files.com...]
Data successfully uploaded!
Command completed successfully.
```

Test Connections

This example shows how to use the tool to test connections to the SFA controllers. If any connection issues are apparent, please correct them prior to collecting data.

```
# ./ddncollect test
Testing connections to controllers.

Continue? [Y]/N :y

Testing connection to SFA0 (user@10.25.80.16):

Password for user@10.25.80.16:
Success!

Testing connection to SFA0 (user@10.25.80.17):

Success!

Testing connection to SFA1 (user@10.25.80.32):

Password for user@10.25.80.32:
Success!

Testing connection to SFA1 (user@10.25.80.33):

Success!

Testing completed. If any connections failed, please correct the configuration and try again.
```

Select Bundle

Sometimes, it is necessary to upload a different bundle other than the one that was just created. Use 'select' to select a different bundle to upload. Note that the tool only looks in the configured output directory for bundles to select.

```
# ./ddncollect select
Bundle(s) currently stored in the specified directory (data):

    [1]. 2021-03-25-15-00-DDN-123456.tar.xz
    [2]. 2021-03-26-14-43-DDN-Test-123456.tar.xz
    [3]. 2021-03-26-13-35-DDN-Test-123456.tar.xz
    [4]. 2021-03-25-14-14-DDN-123456-SFA0-10.36.13.80-diag-pre_service.tar.xz
    [5]. 2021-03-26-13-02-DDN-Test-123456.tar.xz
    [6]. 2021-03-25-14-18-DDN-123456-SFA0-10.36.13.80-diag-pre_service.tar.xz
    [7]. 2021-03-25-14-18-DDN-123456-SFA0-sysinfo.tar.xz
    [8]. 2021-03-25-14-14-DDN-123456-SFA0-sysinfo.tar.xz
    [9]. 2021-03-26-12-18--123456.tar.xz
    [10]. 2021-03-26-12-16--123456.tar.xz
    [11]. 2021-03-26-13-55-DDN-Test-123456.tar.xz
    [12]. 2021-03-25-14-22-DDN-123456.tar.xz

    [M]ain menu

* Bundle currently selected for upload

Enter a number to change selected bundle or 'Q' to quit.
 >> 12

Selected bundle is now: 2021-03-25-14-22-DDN-123456.tar.xz
```

Menu-based Usage

The ddncollect tool provides an easy-to-use interface that guides the user through configuration, collection, and upload through a menuing system. This method is a good choice if you are unfamiliar with the tool.

Menu-based example

The menuing system provides a handy interface for guiding you through using the tool. The main menu is presented here. The status of the tool is shown in bold text at the top, along with the next recommended action. As the user steps through the process, the status changes and the next step is highlighted as well. The actions that are performed when selected are the same ones as using the tool via command-line, so they are not repeated here.

```
# ./ddncollect menu
DDN Support Data Collection Tool (version 1.7)

    STATUS: Tool Installed.    Recommended Action:  2. [C]onfigure the Data Collection Tool

Please select from the following choices. Enter either the number or letter:


    1.  [I]nstall this tool
  * 2.  [C]onfigure the Data Collection Tool settings in ddntools.conf
    3.  [P]assword caching (Status: Not entered)
    4.  [T]est connections to controllers
    5.  [V]iew configuration settings
    6.  [E]xecute data collection according to settings in ddntools.conf
    7.  [S]elect support bundle to upload (Selected: 2021-03-25-14-22-DDN-123456.tar.xz)
    8.  [U]pload selected bundle
    9.  [R]estart SFA API server

    0.  [Q]uit
```

>>

A Note About Passwords

When data is collected, the program attempts to access the controller multiple times in various ways. Each time an access attempt is made, authentication must occur. Depending upon the configurations selected, the runtime may vary from less than one minute up to multiple minutes (if DIAGS are collected). To prevent the user from having to monitor the running program and entering a password for each attempt, the 'user' password is cached in the program memory and subsequently entered automatically for each access attempt. *The cached password is never written to stored disk nor will it appear in any logs or cli commands.*

If DIAGS are collected, password caching is not possible. However, it is possible to not be required to enter them through the use of SSH keys. If you would like to not have the 'diag' user password prompted for input during the execution of the program, you must configure SSH key usage on the controllers or alternatively, configure for SSH multiplexing.

8. Configuration Options

Configuration options are stored within the ddntools.conf file for ease of use and reference. Before collection of data, these options must be set. The options are grouped together into sections regarding user contact information, data collection options, SFA controller(s), and bundle. Each section is described below. You may create your own ddntools.conf either through use of the configure option or directly using the ddntools.conf.template.

ddntools.conf.template

```
#
# Contact Information
#
[Contact]
admin_name =
admin_phone =
admin_email =
company_name =
case_number =

#
# Data Collection Options
#
[Options]
collect_clui = True
collect_diags = False
core_files = 1
directory = /var/tmp/ddncollect
suppress_sfa_logging = True
```

```
use_xz = False
verbose = False
auto_upload = True
# If using proxy, uncomment following line and change server address
#proxy = http://myproxyserver

#
# SFA Controller Config
#
[SFA0]
c0_ip = 10.10.10.1
c1_ip = 10.10.10.2
user_name = user

#
# Optional Additional SFA Controller Config
#
#[SFA1]
#c0_ip = 10.10.10.3
#c1_ip = 10.10.10.4
#user_name = user

#
# Bundle to be uploaded by tool to DDN. Set and used by tool. DO NOT UPDATE.
#
[Bundle]
selected_bundle =

#
# Installation status. Set and used by tool. DO NOT UPDATE.
#
[Install]
install_status = Installed
```

[Contact]

This section allows you to enter the name of the person DDN support should contact regarding SFA operation guidance. Case Number should be the SR number that is produced when the support ticket was opened.

OPTION	ACCEPTED VALUES	DESCRIPTION
admin_name	Alphanumeric text	Name of primary contact of SFA system

OPTION	ACCEPTED VALUES	DESCRIPTION
admin_phone	Alphanumeric text	Phone number where primary contact can be reached
admin_email	Any valid email address	Email address where primary contact can be reached
company_name	Alphanumeric text	Name of your company
case_number	6-digit Support Case Number	Support Case SR number

[Options]

This section allows you to determine which data to collect from the SFA system. There are three ways data is extracted from the system: via API, via CLUI, and via the DIAGS command. API data extraction is always performed. Optionally, CLUI and DIAGS can be collected as well. Please collect what you are directed to by DDN support.

1.

OPTION	ACCEPTED VALUES	DESCRIPTION
collect_diags	True or False	Diagnostic files including vm crash cores
collect_clui	True or False	CLUI output of various commands
core_files	1,2,3,all,lite,report	Number of core files to collect when collecting diags
directory	Any valid directory	Where to place the collected data
suppress_sfa_logging	True or False	If True, informational entries are written to system log
verbose	True or False	If True, details about operations are displayed
auto_upload	True or False	If True, upload to DDN is performed automatically

[SFA0]

Enter the IP addresses of the SFA controllers from which data is to be collected. At least one IP is required (c0_ip). The second controller IP is optional but recommended. If both controllers are accessible, please set both IPs. NOTICE: It does not matter which of these is the PRIMARY controller.

OPTION	ACCEPTED VALUES	DESCRIPTION
c0_ip	IP address or hostname	Controller address or hostname

OPTION	ACCEPTED VALUES	DESCRIPTION
c1_ip	IP address or hostname	Controller address or hostname
user_name	'user' or the configured username	The account name of the controller

[SFA1] through [SFA99]
If multiple SFA systems are to be collected, add a section for each one, containing c0_ip, c1_ip, and user_name for each.

[Bundle]
The bundle section is automatically populated by the tool. The tool populates this option with the file path of the produced bundle, which is not known in advance. If you do need to use the tool to upload a different file, you may choose another one through: *ddncollect select*.

9. What is Collected

Depending on the options selected, various amounts of data are collected from the system. There are three main types of data – API extracted data, CLUI output, and DIAGNOSTIC data. API extracted data is collected every time, while CLUI and DIAG data are optional. The API extracted data consists of system configuration, logs, and the contact information provided by the user of the tool. The contact information is stored in the BundleInfo.json exclusively. This file also contains important details about the SFA system such as the model, controller serial numbers, and license information.

Each collection is bundled separately as it is collected, then bundled and compressed together as a final step. If DIAGS and CLUI are collected from both controllers, a maximum of five file will exist in the final bundle. For example:

```
# tar -Jtvf 2021-03-26-14-43-DDN-Test-123456.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin  4161 Mar 25 12:59 logs/ddncollect-install-2021.03.25.12.59.28.log
-rw-r--r--  0 sfaadmin sfaadmin 20550 Mar 25 14:36 logs/2021-03-25-14-22-ddncollect-runtime.log
-rw-r--r--  0 sfaadmin sfaadmin   845 Mar 26 13:10 logs/2021-03-26-13-10-ddncollect-error-stacktrace.log
-rw-r--r--  0 sfaadmin sfaadmin 20622 Mar 25 13:22 logs/2021-03-25-13-22-ddncollect-runtime.log
-rw-r--r--  0 sfaadmin sfaadmin 74656 Mar 26 14:44 SFA0/2021-03-26-14-43-DDN-Test-123456-SFA0-
sysinfo.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin 187784 Mar 26 14:45 SFA1/2021-03-26-14-43-DDN-Test-123456-SFA1-
sysinfo.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin 223716 Mar 26 14:45 SFA3/2021-03-26-14-43-DDN-Test-123456-SFA3-
sysinfo.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin 4143749 Mar 26 14:46 SFA0/2021-03-26-14-43-DDN-Test-10.36.13.80-123456-
SFA0-text-pre_service.txt
-rw-r--r--  0 sfaadmin sfaadmin 4143722 Mar 26 14:46 SFA0/2021-03-26-14-43-DDN-Test-10.36.13.81-123456-
SFA0-text-pre_service.txt
-rw-r--r--  0 sfaadmin sfaadmin 16074599 Mar 26 14:46 SFA1/2021-03-26-14-43-DDN-Test-10.36.13.45-123456-
SFA1-text-pre_service.txt
-rw-r--r--  0 sfaadmin sfaadmin 16074604 Mar 26 14:46 SFA1/2021-03-26-14-43-DDN-Test-10.36.13.46-123456-
SFA1-text-pre_service.txt
```

```
-rw-r--r--  0 sfaadmin sfaadmin 17854989 Mar 26 14:47 SFA3/2021-03-26-14-43-DDN-Test-10.36.59.114-123456-
SFA3-text-pre_service.txt
-rw-r--r--  0 sfaadmin sfaadmin 17854967 Mar 26 14:47 SFA3/2021-03-26-14-43-DDN-Test-10.36.59.115-123456-
SFA3-text-pre_service.txt
-rw-r--r--  0 sfaadmin sfaadmin    91048 Mar 26 14:47 SFA0/2021-03-26-14-43-DDN-Test-123456-SFA0-
10.36.13.80-diag-pre_service.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin    91356 Mar 26 14:48 SFA0/2021-03-26-14-43-DDN-Test-123456-SFA0-
10.36.13.81-diag-pre_service.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin   282820 Mar 26 14:49 SFA1/2021-03-26-14-43-DDN-Test-123456-SFA1-
10.36.13.45-diag-pre_service.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin   283120 Mar 26 14:50 SFA1/2021-03-26-14-43-DDN-Test-123456-SFA1-
10.36.13.46-diag-pre_service.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin   665008 Mar 26 14:56 SFA3/2021-03-26-14-43-DDN-Test-123456-SFA3-
10.36.59.114-diag-pre_service.tar.xz
-rw-r--r--  0 sfaadmin sfaadmin   676652 Mar 26 15:00 SFA3/2021-03-26-14-43-DDN-Test-123456-SFA3-
10.36.59.115-diag-pre_service.tar.xz
```

The final bundle naming format includes your company name, SFA platform, controller serial numbers, and case number. Its format is as follows: YYYY-MM-DD-HH-mm-CompanyName-SystemPlatform-Controller0Serial-Controller1Serial-CaseNumber.tar

10. System API Data

Data collected via API is system health status, configuration details, and the system logs. The data is stored in JSON files and bundled into a single file named YYYY-MM-DD-HH-mm-CompanyName-SystemPlatform-sysinfo.tar. This data is always collected and there is no option to not collect it.

The following tables show potentially sensitive data included in the files.

IP Addresses

These are the IP addresses collected via the SFA API and the file(s) they are found in:

FILE	FIELD	DESCRIPTION
SFAController.json	NTPSyncAddress	IP Address of the NTP server
SFADiscoveredInitiator.json	IPAddress	IP Address of the Initiator
SFAEnclosure.json	BmcIPAddress	BMC IP Address
	BmcIPAddressAlt	BMC Alternate IP Address
SFAHostChannel.json	IPAddress	Host Channel IP Address
SFAICLChannel.json	EthernetIPAddress	ICL Channel IP Address

FILE	FIELD	DESCRIPTION
SFASNMPTrapAgent.json	IPAddress	SNMP Trap Agent IP Address
SFAUserInterface.json	SyslogIPAddress	Syslog IP Address
	SNMPTrapIPAddress	SNMP Trap IP Address
	EmailIPAddress	Email server IP Address
	EmailBackupIPAddress	Email server Backup IP Address
	IPv4Addresses	Controller IPv4 Address
	IPv6Addresses	Controller IPv6 Address
	ControllerIPAddress	Controller Alternate IP Address
	ControllerIPGateway	Controller Gateway IP Address
	ControllerIPGatewayAlt	Controller Gateway Alternate IP Address

Email Addresses

These are the Email Addresses collected via the SFA API and the file(s) they are found in:

FILE	FIELD	DESCRIPTION
SFAUserInterface.json	EmailToAddress	Email To Address
	EmailFromAddress	Email From Address

Hostnames

These are the Hostnames collected via the SFA API and the file(s) they are found in:

FILE	FIELD	DESCRIPTION
SFAStorageSystem.json	Name	System hostname (not canonical)
SFAController.json	Name	Controller hostname (not canonical)
	FWBuildHost	Firmware Build hostname (not canonical)

Usernames

These are the Usernames collected via the SFA API and the file(s) they are found in:

FILE	FIELD	DESCRIPTION
SFAUserAccount.json	Name	Controller user account username
	GroupName	Controller user account group
	UserID	Controller account UID
	GroupID	Controller account GID

BundleInfo.json

This is an example of what is stored in the BundleInfo.json file:


```
[
 {
  "Customer": "DDN",
  "ContactPhone": "555-555-5555",
  "uuid": "60001ff0a0af60000000000030000000",
  "DateAndTime": "2020-03-26T10:36:49.922987-06:00",
  "Controller0Serial": "NT7F12061-R",
  "LicenseNeeded": true,
  "ContactEmail": "support@ddn.com",
  "Platform": "SFA7700X",
  "Controller1Serial": "null",
  "ContactName": "Mr. S. Fusion Administrator",
  "TimeZone": "America/Denver",
  "Licenses": [],
  "Description": "1-SS7700X(head only) with 0 missing enclosures (AUTOMATIC Selection).",
  "CollectorVersion": "1.0",
  "SFAAPIVersion": "11.7.0",
  "CaseNumber": "00508"
 }
]
```


Sample 2020-02-25-11-12-My-Company-SFA7700XE-sysinfo.tar

```
# ll
total 4676
-rw-r--r--. 1 root root     623 Feb 25 13:10 BundleInfo.json
-rw-r--r--. 1 root root 3584272 Feb 25 13:10 logmessages0.txt
```

```
-rw-r--r--. 1 root root   77397 Feb 25 13:10 logmessages1.txt
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAAllowedIPRange.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAAuthentication.json
-rw-r--r--. 1 root root    4103 Feb 25 13:10 SFAClientIOC.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAConfigurationFile.json
-rw-r--r--. 1 root root    9848 Feb 25 13:10 SFAConnector.json
-rw-r--r--. 1 root root    3293 Feb 25 13:10 SFAController.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFADiscoveredInitiator.json
-rw-r--r--. 1 root root    5762 Feb 25 13:10 SFADiskChannel.json
-rw-r--r--. 1 root root  272837 Feb 25 13:10 SFADiskDrive.json
-rw-r--r--. 1 root root  251802 Feb 25 13:10 SFADiskDriveStatistics.json
-rw-r--r--. 1 root root   88814 Feb 25 13:10 SFADiskSlot.json
-rw-r--r--. 1 root root     716 Feb 25 13:10 SFAEnclosureConfigurations.json
-rw-r--r--. 1 root root    7891 Feb 25 13:10 SFAEnclosure.json
-rw-r--r--. 1 root root    7834 Feb 25 13:10 SFAExpander.json
-rw-r--r--. 1 root root    7020 Feb 25 13:10 SFAFan.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAHostChannel.json
-rw-r--r--. 1 root root     810 Feb 25 13:10 SFAHost.json
-rw-r--r--. 1 root root    6884 Feb 25 13:10 SFAICLChannel.json
-rw-r--r--. 1 root root    2368 Feb 25 13:10 SFAICLIOC.json
-rw-r--r--. 1 root root     738 Feb 25 13:10 SFAImage.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAInitiator.json
-rw-r--r--. 1 root root    3946 Feb 25 13:10 SFAInternalDiskDrive.json
-rw-r--r--. 1 root root    3518 Feb 25 13:10 SFAIOC.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAJob.json
-rw-r--r--. 1 root root    5486 Feb 25 13:10 SFAPowerSupply.json
-rw-r--r--. 1 root root   16370 Feb 25 13:10 SFAPresentation.json
-rw-r--r--. 1 root root     960 Feb 25 13:10 SFARAIDProcessor.json
-rw-r--r--. 1 root root    2510 Feb 25 13:10 SFASEP.json
-rw-r--r--. 1 root root     823 Feb 25 13:10 SFASNMPAgent.json
-rw-r--r--. 1 root root     435 Feb 25 13:10 SFASNMPTrapAgent.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFASNMPTrapReceiver.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFASNMPUser.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFASparePool.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFAStack.json
-rw-r--r--. 1 root root   23356 Feb 25 13:10 SFAStoragePool.json
-rw-r--r--. 1 root root    1580 Feb 25 13:10 SFAStorageSystem.json
-rw-r--r--. 1 root root       2 Feb 25 13:10 SFASyslogAgent.json
-rw-r--r--. 1 root root   19223 Feb 25 13:10 SFATemperatureSensor.json
-rw-r--r--. 1 root root     314 Feb 25 13:10 SFAUnassignedPool.json
-rw-r--r--. 1 root root    2358 Feb 25 13:10 SFAUPS.json
-rw-r--r--. 1 root root    1229 Feb 25 13:10 SFAUserAccount.json
-rw-r--r--. 1 root root     357 Feb 25 13:10 SFAUserGroup.json
-rw-r--r--. 1 root root    1379 Feb 25 13:10 SFAUserInterface.json
-rw-r--r--. 1 root root   29088 Feb 25 13:10 SFAVirtualDisk.json
-rw-r--r--. 1 root root  157540 Feb 25 13:10 SFAVirtualMachine.json
-rw-r--r--. 1 root root     902 Feb 25 13:10 SFAVirtualProcessor.json
```

```
-rw-r--r--. 1 root root   48049 Feb 25 13:10 SFAVoltageSensor.json
-rw-r--r--. 1 root root      32 Feb 25 13:10 UUID
```

11. CLUI Data

Textual output of SFA CLUI commands is collected if this option is selected. It is stored in a bundle named YYY-MM-DD-HH-MM-CompanyName-controllerIndex-text-pre_service.txt. The command list is as follows:

```
show subsystem summary
show subsystem summary all
show subsystem fault
show subsystem fault all
app show subsystem summary
app show subsystem summary all
app show channel errors
show sas_connector
show sas_connector all
show ioc fault
show ioc phy
show physical_disk phy_errors
ui show api
ui show account
ui show account all
ui show cli
ui show email
ui show gui
ui show network
ui show network all
ui show ntp
ui show snmp_agent
ui show snmp_trap_agent
ui show stats_server
ui show syslog
```

Sample CLUI file: 2020-02-25-11-12-My-Company-c0-text-pre_service.txt

```
# head -25 2020-02-25-11-12-My-Company-c0-text-pre_service.txt
Keyword belongs to a mutually exclusive group that has already been specified,
please choose from:
Mutually Exclusive Optional:
Optional keywords:
   ALL_ATTRIBUTES   Show all attributes for the specified object

Tue Feb 25 11:12:58 2020

****************************
*     Subsystem Summary     *
```

```
*****************************


*********************
*     Subsystem     *
*********************


                                                                  |   Locate   |   Fast   |  Verify
Name                 |Time                       | Licenses | Dwell Time | Timeout |   Policy   | UUID
| Platform   |
-------------------------------------------------------------------------------------------------------------
----------------------------------
es7k01                Tue Feb 25 11:12:58 2020             120 seconds    ON      DISABLED
60001ff0a11e60000000000030000000  SFA7700XE


*************************
```

12. Diagnostic Data

If the DIAG option is selected, a collection of files containing developer logs, kernel logs, and system crash core files is produced. Developer logs contain state of SFA OS firmware including memory contents, including a very high resolution log file called logdisk for SFA developer analysis. Kernel logs contain syslog, dmesg, log files from /var/log, and other files. During unexpected firmware events (e.g. null pointer access) core files saved in /tmp/corefiles for each RP (for 14KX, 2 core files) as janus_core-n.m “n” is the RP number, starting with 0; “m” is the corefile number, 1 for most recent, 2 for 2nd most recent crashes. Only last 4 instances kept.

It is stored in a bundle named YYY-MM-DD-HH-MM-CompanyName-controllerIndex-diag-pre_service.xz.

Sample DIAG file: 2020-02-25-11-12-My-Company-c0-diag-pre_service.xz

The file tree of the contents of this bundle is shown here:

```
[root@vm78-centos7 data]# tar -Jxvf 2020-02-25-11-12-My-Company-c0-diag-
pre_service.xz
tmp/audit
ddn/janus_version.txt
boot/grub/
boot/grub/interfaces.4
boot/grub/lost+found/
boot/grub/interfaces.1
boot/grub/interfaces.2
boot/grub/config/
boot/grub/config/snmpd.conf
boot/grub/config/gui.conf
boot/grub/config/timezone.conf
boot/grub/config/snmp_trap.conf
boot/grub/config/etc/
```

```
boot/grub/config/etc/shadow
boot/grub/config/smtp.conf
boot/grub/config/syslog.conf
boot/grub/config/stats.conf
boot/grub/config/roles.conf
boot/grub/config/snmp/
boot/grub/config/snmp/mibs/
boot/grub/config/snmp/mibs/SFA-TRAP-MIB.txt
boot/grub/config/snmp/mibs/.index
boot/grub/config/snmp/mibs/SFA-INFO.txt
boot/grub/config/cpPersStoreFile.db
boot/grub/config/api.conf
boot/grub/interfaces.tmp
boot/grub/menu.lst
boot/grub/interfaces
boot/grub/stage1
boot/grub/interfaces.3
boot/grub/stage2
boot/grub/default
boot/grub/menu.lst.template
boot/grub/e2fs_stage1_5
boot/grub/uuid.txt
tmp/diag_health_monitoring_report.txt
tmp/ipmitool_sel_list.txt
tmp/ipmitool_sel_list.raw
tmp/systemd_journal.txt.gz
tmp/dmesg.current
var/log/auth.log
var/log/auth.log.1
var/log/auth.log.2.gz
var/log/auth.log.3.gz
var/log/auth.log.4.gz
var/log/boot.log
var/log/boot.log.0
var/log/boot.log.10.gz
var/log/boot.log.11.gz
var/log/boot.log.12
var/log/boot.log.1.gz
var/log/boot.log.2.gz
var/log/boot.log.3.gz
var/log/boot.log.4.gz
var/log/boot.log.5.gz
var/log/boot.log.6.gz
var/log/boot.log.7.gz
var/log/boot.log.8.gz
var/log/boot.log.9.gz
var/log/janus_config.log
```

```
var/log/janus_config.log.1
var/log/janus_config.log.2
var/log/janus_config.log.3
var/log/janus_config.log.4
var/log/janus_config.log.5
var/log/janus_config.log.6
var/log/janus_config.log.7
var/log/jerry.log
var/log/janus/
var/log/janus/eventlog
var/log/syslog
var/log/syslog.1
var/log/syslog.2.gz
var/log/syslog.3.gz
var/log/syslog.4.gz
var/log/syslog.5.gz
var/log/syslog.6.gz
var/log/syslog.7.gz
tmp/corefiles/compressed-persist-data-file
tmp/corefiles/cpld.log
tmp/corefiles/cpld.log.1
tmp/corefiles/cpld.log.10
tmp/corefiles/cpld.log.2
tmp/corefiles/cpld.log.3
tmp/corefiles/cpld.log.4
tmp/corefiles/cpld.log.5
tmp/corefiles/cpld.log.6
tmp/corefiles/cpld.log.7
tmp/corefiles/cpld.log.8
tmp/corefiles/cpld.log.9
tmp/corefiles/cpPersStoreFile.db1442031319 (2015-09-11 20:15:19)
tmp/corefiles/cpPersStoreFile.db1467393207 (2016-07-01 10:13:27)
tmp/corefiles/dmesg.txt.1
tmp/corefiles/persistent-data-on-quorum
tmp/corefiles/Stack-0x10000000.log
tmp/corefiles/Stack-0x10000000.log.1
tmp/corefiles/Stack-0x10000000.log.10
tmp/corefiles/Stack-0x10000000.log.2
tmp/corefiles/Stack-0x10000000.log.3
tmp/corefiles/Stack-0x10000000.log.4
tmp/corefiles/Stack-0x10000000.log.5
tmp/corefiles/Stack-0x10000000.log.6
tmp/corefiles/Stack-0x10000000.log.7
tmp/corefiles/Stack-0x10000000.log.8
tmp/corefiles/Stack-0x10000000.log.9
tmp/corefiles/Stack-0x10000001.log
tmp/corefiles/Stack-0x10000001.log.1
```

```
tmp/corefiles/Stack-0x10000001.log.2
tmp/corefiles/Stack-0x10000001.log.3
tmp/corefiles/Stack-0x10000001.log.4
tmp/corefiles/Stack-0x10000001.log.5
tmp/corefiles/Stack-0x10000001.log.6
tmp/corefiles/Stack-0x10000001.log.7
tmp/corefiles/Stack-0x10000001.log.8
tmp/corefiles/Stack-0x10000001.log.9
tmp/corefiles/testfile
```

13. What is Not Collected

Only system data and logs are collected. Absolutely no data coming from file systems or the data stored on the SFA systems is gathered. This includes file system structure and format, including directory paths, mount points, or file names.

14. Uninstall

To uninstall the tool, simply delete all the files stored in the installation directory. Optionally, you may also delete any prerequisite packages that were required for the install. No additional files are installed or changed outside of the installation directory.

15. Support

If you have questions or require assistance, contact DDN Support:

Web

DDN Community Support Portal	https://community.ddn.com/login
Portal Assistance	webportal.support@ddn.com

Telephone

DDN Support Worldwide Directory	https://www.ddn.com/support/global-services-overview/

Email

Support Email	support@ddn.com

Bulletins & Notices

Support Bulletins	http://www.ddn.com/support/technical-support-bulletins
End-of-Life Notices	http://www.ddn.com/support/end-of-life-notices
Release Notes	https://community.ddn.com/login
Subscription Requests	support-tsb@ddn.com