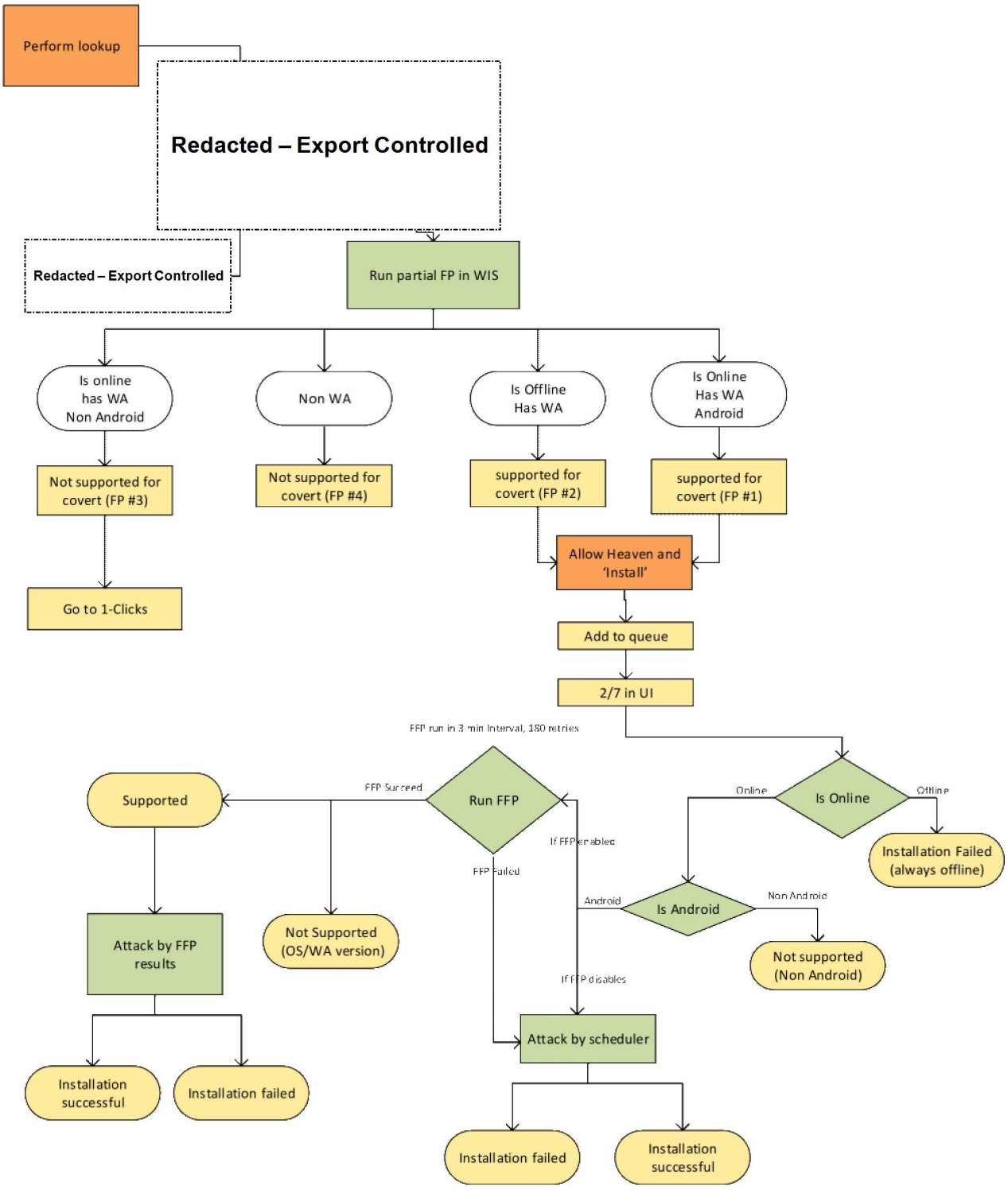
Exhibit 38

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

Hummingbird/Heaven

Tuesday, July 10, 2018 1:23 PM



Configuration log

Heaven installation flow may change from time to time, this log will try to keep up with these changes:

10-July-2018 : FullFingerPrint is not in use

Exhibit 39

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

Heaven - Full PRD

Created by [redacted] last modified on Jul 25, 2018

Doc version: Heaven PRD A.5 18.07.docx

Product Requirements Document	
Pegasus 2 – Heaven	
Document Identification	
Release Version:	
Distribution Status:	For Internal Use Only
Revision No.	
Issue Date:	27/12/2017
Author	Dana Bargury & Kfir Fleischer

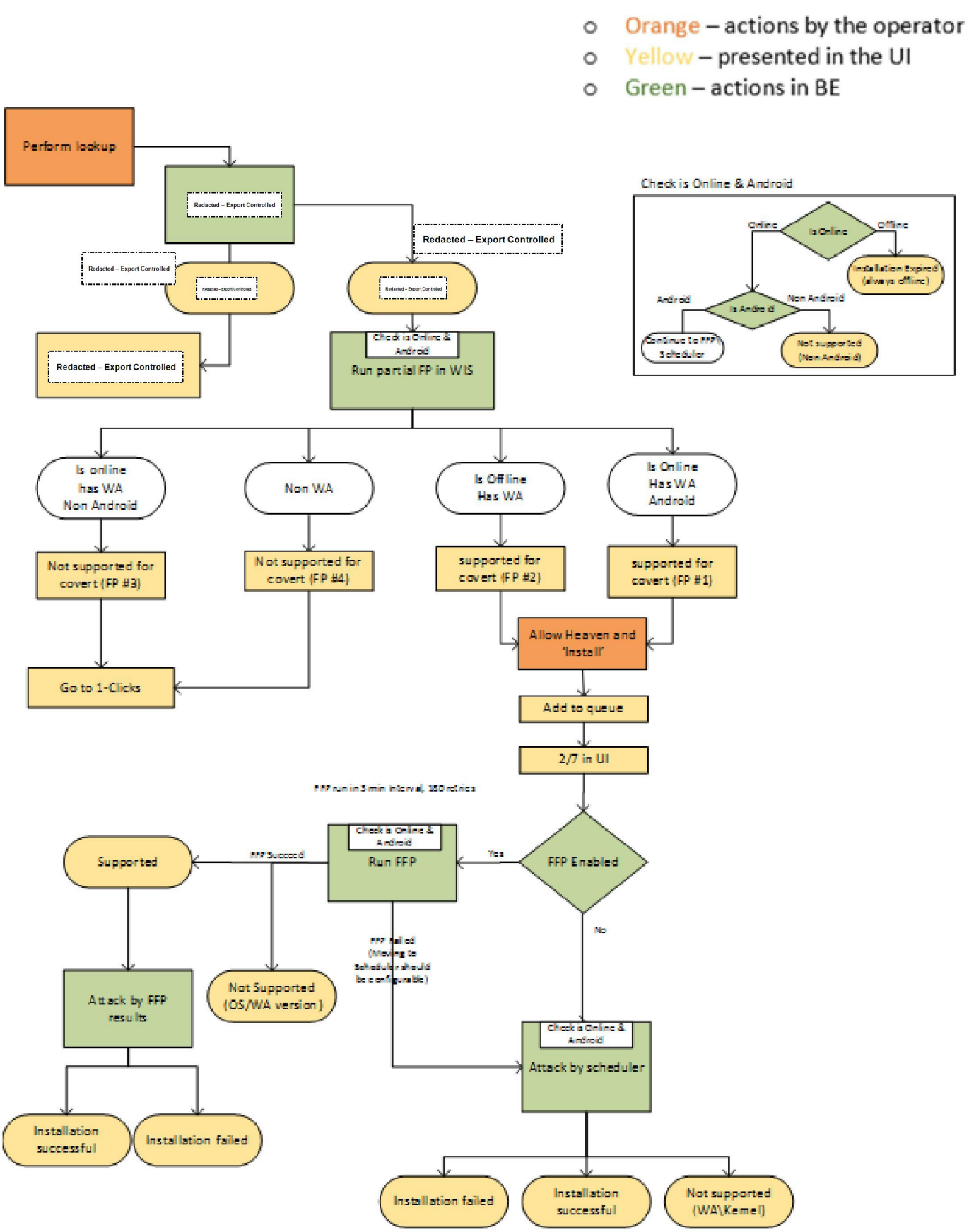
Revision History

Revision	Document Status	Author	Date
R0	Draft	Dana Bargury & Kfir Fleischer	27/12/17
R1		Dana Bargury	17/1/18
R2		Moran Shohat	09/04/18
R3		Moran Shohat	11/06/18
R4		Moran Shohat	18/06/18
R5		Moran Shohat	18/07/18

Contents

1 Workflow & Requirements

Installation flow



Requirements

Requirement H20000010 – Initial lookup \ Partial FP

- **Redacted – Export Controlled**

Redacted – Export Controlled

- WIS shall perform a partial FP, which indicates if the target is candidate for covert installation. The candidates are type 1 'Is online + Android +has WA' and type 2 'Is offline + has WA' (see installation flow diagram).
- Partial FP responses should be cached for 30 min (configurable) [DB1] [MS2] [MS3] [MS4]
- There is no limitation on the number of partial FP per time interval.

PF #	PF Results	Message	Vectors
1	is Online + has WA + Android	Target device is Android and a candidate for Covert Android	CovertAndroid
			CraftedSMS
			OTA
2	is Offline + has WA	Target device is unknown but there is potential for Covert Android	CovertAndroid
			CraftedSMS
			OTA
3	Is Online + has WA + Non Android	Target device is not Android and is not supported for Covert Android	CraftedSMS
			OTA
4	Non WA	Target device is not supported for covert installations	CraftedSMS
			OTA
5	Error	Fingerprint service returned an error	CraftedSMS
			OTA

Here is the table to include the responses and vectors according to FP:

Requirement H20000020 – Installation vector

- If the partial FP response identified that the target "is offline and has WhatsApp" or the target "is online and has WhatsApp and Android", the operator should be able to choose the installation vector: CovertAndroid
  - Note that, 'is offline + has WA' is a candidate because there is no way to be sure it's not an Android device in this stage. In the next steps of the installation flow it will be checked again (not android device won't be attacked).
  - The CraftedSMS\OTA options should also be allowed

Requirement H20000030 – Operator sending the installation

- In the installation phase 2 in the UI –
  - The domain should be chosen by default per the installation vector – separate for Heaven installations
  - The user will not be required to add a redirect link
  - The user will click “install”

Requirement H20000040 – Installation added to queue

After the user had clicked the install bottom, the UI log should be: “Installation added to queue” and the UI status will change to 2/7 and the operator can’t abort

Requirement H20000050 – Daily limit

- [24H/Cool down period* Number of attacks accounts] Heaven installations are allowed per day (local time)
- For detailed limitations table go to 'Heaven system architecture' (here)

	Topic	Interval	Notes
1	**Maximum installations attempts per day**	[24H/Cool down period* Number of attacks accounts]	
2	**Maximum attempts per phone number**	2 exploit failures	
3	**Max number of Install retries due to "comm failure" / time between retries**	• Max 20 attempts • 5 minute between attempts • Limit applies to both Immediate and Scheduler Installs	
4	**Max number of Install retries due to "offline" / time between retries**	• Max 20 attempts • 5 minute between attempts • Limit applies to both Immediate and Scheduler Installs	

5	**Max number of Install retries due to "exploit failure" / time between retries**	• Max 2 attempts • 5 minute between attempts • Target receive WhatsApp call on each attempt • Limit applies to both Immediate and Scheduler Installs	
6	**Max number of Install retries due to "Unexpected error" / time between retries**	• Max 2 attempts • 5 minute between attempts • Target might receive WhatsApp call on each attempt • Limit applies to both Immediate and Scheduler Installs	
8	**Time to wait if target has no internet**	5 minute	Up to 20 retries before failing the installation
9	**Cooldown period between heaven failure and 1 click**	3 Days	• Enforced by UI in Pegasus 2 • Enforced by Server in Pegasus 3
10	**Time between full fingerprint attempts**	3 minutes	
11	**Max number of full fingerprint attempts per target**	180	

Requirement H20000060 – Full FP & Attack

- 1st stage:
  - Check if the target is online or offline, the target should be in Idle stage (Offline for 18 min) in order to perform FFP.
  - Check if the target is Android
- 2nd stage:
  - Run FFP
    - If the **target is supported** – go to sending the payload by the exact parameters
      - Presents FFP results in installed log in installation tab in the UI. Add **'Device details'** to the list to present FFP result as **"Android 5.0, SM-G900F, App V2.18.79"**

      If the **target is not supported** ('OS/WA version is not supported') –
        - Fail the installation, with the log : "Installation failed: Device not supported for Covert Android"
        - ~~Add the target phone number to a blacklist for heaven installations~~ (should be removed from 2.70)
        - If the **target is not supported** (Non Android) – Fail the installation, with the log: "Installation failed: Target device is not Android. "
- 3rd Stage: **FFP Disabled\ FFP data wasn't received\ FFP failed**
  - Go to payload sending by the scheduler
  - The ability of moving to scheduler should be configurable
  - If the target is offline for X retries the installation will failed.
    - If the **target was offline**-
      - Expired the installation, with the log: "Installation expired: Target device was offline"
- If the **target is not supported** (Non Android) – Fail the installation, with the log: "Installation failed: Target device is not Android. "
- If the **target is not supported** (WA version is not supported) – Fail the installation, with the log: "Installation Failed: Device is not supported"
- This will occur only after the first call made when the WIS recognized the WA is not supported and send the relevant status code to the PS.

Requirement H20000070 – Managing the WhatsApp users list

- WhatsApp users - credentials from whitened SIM imported from android device
- Need to hold 2 lists:
  - Users for FP
  - Users for payloads sending
  - Each list needs to contains at least X users [DB5]
  - Choosing the user per target is:
    - Redacted – Per Court Order
    - User should be active (detailed requirements here)
    - Consistent - once a WA user was chosen for the target it will not change unless the user is blocked, if the user is blocked chose a different one.
    - Once user is blocked there should be an alert (for Nagios)
    - Once all users are blocked there should be an alert (for Nagios)
    - For payload sending users - each user can be used only for 1 target per day
    - Expiration: the user will expire after 2 months

Requirement H20000071 - Blocked WA accounts

- WIS should send to the PS specific status code when the account is blocked
- PS should mark the specific WA Credentials as blocked with timestamp when it happened
- PS should not use blocked WA Credentials
- PS should alert when the WA Credentials is blocked with indication which WA Credentials was blocked and which Request Type (FP/Attack) was requested before blocking for Nagios monitoring
- PS should alert when there are no WA Credentials available for Nagios monitoring

Requirement H20000072 - WA Accounts availability

1. After pressing 'Install', check for at least 'MIN FP accounts' (should be considered if FFP is enabled) and 'MIN Attack accounts' available, i.e., not blocked or expired.
2. After FFP succeed/ FFP Failed/ FFP disabled (scheduler stage)

1. If there is → continue to installation
2. Else →
    - UI pop up "Covert Android service is currently unavailable. Please contact your system administrator"
    - Add indicative log in the DB

1. Min FP accounts – X
2. Min Attack account available – Y
3. Support X=0 or Y=0 in the configuration (0 in configuration doesn't disable the feature, rather allows to not set a minimum number and install won't be possible to sent)

1. if all attack accounts are in cool down -> continue the installation until 'installation overall time expired' (check again in X min)
    - in PS2 : Agreed it will be the same mechanisms of installation expiration today (1440 sec for overall installations)
2. if there were no accounts available during the attack stage time →installation should be failed
    - There is no option to add indicative log, so we are assuming that Heaven installation that failed due to expiration limit will be because the attack accounts were unavailable.

Requirement H20000080 – Payload sending

- The payload should sent only to Android devices
- By scheduler:
    - Start with latest WA version
    - Interval: X min
    - Up to 2 payloads per target
    - By exact parameters (from full FP):
        - Send 1 payload of the exact version
        - If the installation fails:
            - Fail the installation with the UI log: " XXX"
            - ~~Add the number to the blacklist~~ (should be removed from 2.70)

Note that, retries mechanism of communication, etc is explained in 'Heaven system architecture' (here) and in **Requirement H20000050**

Requirement H2000090 – Cleaning traces

- The agent should clean traces from the device:
    - Number sanitizing [redacted]
    - Clean the crashes created on the device (added [DB10] in 2.66)

Requirement H20000100 – Abuse prevention

- According to abuse prevention table for 2.70 version (here)

2 Deployment requirements

1. 2 VPS servers for each client:
    a. 1 for FP WIS clients
    b. 1 for attack WIS clients
    c. 8 whitened VPS servers for each client:
        i. 3 VPS servers for FP-server
        ii. 3 for attack-server
        iii. 1 for relay-server
        iv. 1 for honeypot
        v. 14 local SIMs:
            1. 7 for FP-WIS-clients
            2. 7 for attack-WIS-clients

3 BI

TBD – joint work with Moran

4 Future enhancement

TBD

5 Compliance Matrix

Table 1 concludes the requirements described in this document and indicated the current status & Open issues

Table 1: Compliance Matrix

#	Requirement	Status	Open issues
1.	Requirement H20000010 – Initial lookup Partial FP		
1.	Requirement H20000020 – Installation vector		
1.	Requirement H20000030 – Operator sending the installation		
1.	Requirement H20000040 – Installation added to queue		
1.	Requirement H20000050 – Daily limit		
1.	Requirement H20000060 – Full FP		
1.	Requirement H20000070 – Managing the WhatsApp users list		
1.	Requirement H20000071 – Blocked WA accounts		
1.	Requirement H20000072 – WA Accounts availability		
1.	Requirement H20000080 – Payload sending		
1.	Requirement H2000090 – Cleaning traces		
1.	Requirement H20000100 – Abuse prevention	Exists	

No labels

Exhibit 40

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

Pegasus 3 - WIS Refactoring (Heaven Separati

Created by [redacted] on Apr 29, 2018

You are viewing an old version of this page. View the current version.

Compare with Current · Restore this Version · View Page History

Version 1 Next »

First draft base on Alex's work, unreviewed.

1. Background

Due to the relatively high rate of change in Heaven's implementation, there is a need to decouple the business logic from Pegasus 3, so that updates can be released without requiring a full Pegasus 3 backend release.

Another way of stating this is that the design principle of "WIS is stateless" is being changed in order to shorten the release cycle.

2. Overview

1. Heaven-specific business logic shall move from Pegasus 3 server to "enhanced" WIS, to be called Android Backend Server (ABS).
2. ABS is responsible for the "business logic" from fingerprint (partial and full) to PE installation. Uploading the agent remains Pegasus server's responsibility.
3. In addition to the existing /fingerprint and /install RESTful API from Pegasus 3 to ABS, new APIs will be defined from ABS to Pegasus 3, as described below.
4. The following diagrams provides an overview of the solution

TBD (revised diagram from Alex)

TBD (Sequence diagram)

3. Details

3.1. Roles and responsibilities (R&R)

3.1.1. Pegasus 3 Server

1. Validating customer's licenses
2. Creating installation attempt with unique identifier (iKey).
3. Sending installation request to ABS.
4. Validating installation state for current installation attempt.
5. Updating delayed installation expiration.
6. Managing packages and providing zip with required payload (exploits, PE, etc.) on demand.
7. Performing GeoIp validation.

8. Performing certificate digest validation.
9. Providing Agent bundle.

3.1.2. ABS

1. Managing WhatsApp accounts.
2. Managing Heaven configurations.
3. Performing full fingerprint logic.
4. Scheduling Heaven attacks.
5. Providing exploit, pre-pe and PE.
6. Preserving fingerprint and attack results for BI purposes.
7. Preserving Heaven installation state to ensure persistency.
8. Operating state machine for relevant Heaven steps.

3.2. API

3.2.1. ABS API

3.2.1.1. /fingerprint

Instruct ABS to perform partial fingerprint on speficied target

Parameters:

- Target phone number in plain text

3.2.1.2. /install

Instruct ABS to perform an installation on specified target. Full fingerprint will be performed first (if enabled), followed by attack (scheduled or direct).

Parameters:

- Target phone number in plain text
- Unique installation identifier (iKey)
- Zip with all required payload files (exploit, stager, pre-pe, pe).

3.2.2. Pegasus 3 Server API

3.2.2.1. /prepareAttack

Pegasus will perform the following upon receipt of this API

1. Verifying installation state validity.
2. Setting delayed installation expiration.
3. Providing zip with all required payload files (exploit – for specific versions, stager, pre-pe, pe).
4. Store received RC4 key for agent encryption.

Parameters:

- Unique installation identifier (iKey).
- WhatsApp version.
- Android version.
- Device model.
- RC4 key.

3.2.2.2. /installResult

This API is used to indicate failure of a Heaven attack. In case of success, the target device will contact Pegasus for agent download.

Parameters:

- Unique installation identifier (iKey).

Returned parameters:

- Status code
- Reason code

Roles and responsibilities (R&R):

a. IS/PS will be responsible for:

i. Validating customer's licenses.
ii. Creating installation attempt with unique identifier (iKey).
iii. Sending installation request to ABS.
iv. Validating installation state for current installation attempt.
v. Updating delayed installation expiration.
vi. Managing packages and providing zip with required payload (exploits, PE, etc.) on demand.
vii. Performing GeoIp validation.
viii. Performing certificate digest validation.
ix. Providing Agent bundle.

b. ABS will be responsible for:

i. Managing WhatsApp accounts.
ii. Managing Heaven configurations.
iii. Performing full fingerprint logic.
iv. Scheduling Heaven attacks.
v. Providing exploit, pre-pe and PE.
vi. Preserving fingerprint and attack results for BI purposes.
vii. Preserving Heaven installation state to ensure persistency.
viii. Operating state machine for relevant Heaven steps.

High-level APIs:

a. ABS Heaven APIs:

i. /fingerprint :
1. ABS action - partial fingerprint.
2. Parameters:
a. Target phone number in plain text

ii. /install :

1. ABS action – full fingerprint (if turned on in configuration) followed by attack (either scheduled or direct).
2. Parameters:
    a. Target phone number in plain text
    b. Unique installation identifier (iKey)
    c. Zip with all required payload files (exploit, stager, pre-pe, pe).

b. PS Heaven APIs:

i. API before each attack try:

1. PS actions:
    a. Verifying installation state validity.
    b. Setting delayed installation expiration.
    c. Providing zip with all required payload files (exploit – for specific versions, stager, pre-pe, pe).
    d. Persisting received RC4 key for agent encryption.
2. Parameters:
    a. Unique installation identifier (iKey).
    b. WhatsAp version.
    c. Android version.
    d. Device model.
    e. RC4 key.

ii. API for install result:

1. PS action – in case of successful installation, no result is required as device browsing PS will speak for themselves. However it is important in case of final failure.
2. Parameters:
    a. Unique installation identifier (iKey).
    b. Status code.
    c. Reason code.

No labels

Exhibit 41

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

Heaven System Architecture Specification - DRAFT

Created by [redacted] last modified on Nov 28, 2017

You are viewing an old version of this page. View the current version.

Compare with Current · Restore this Version · View Page History

« Previous **Version 8** Next »

Initial draft - Not reviewed

1. General

1.1. Background

"Heaven" is a zero-click installation process based on vulnerabilities in WhatsApp, initially implemented for Android phones.

1.2. Solution Overview

Components:

1. **WIS** (WhatsApp Installation Server) - a per-customer stateless server interfacing with IS over RESTful API, serving Fingerprint and Install requests as described below
2. **Fingerprint Server** - a globally unique NSO server needed for the fingerprint process
3. **Relay** - a per-customer UDP server for WhatsApp signalling that is used as part of the attack
4. **IS/PS** - As usual

1.3. Scope

This solution applies to all Android phones running Lolipop or later (specifically, using jmalloc), with WhatsApp installed and registered.

1.4. Limitations

1. Target must have WhatsApp
2. Currently only support for WhatsApp version 395
3. 64bit with JEmalloc
4. 32bit with JEmalloc (statistical success)
5. Internet connection
6. Exposure:
    a. Target phone will ring with WhatsApp voice call for 1-2 seconds
    b. WhatsApp crashes if disconnected during attack, worse case will continuously crash until phone restarted.
7. ROP must be updated per WhatsApp update (every 3 weeks)
8. Flow will need to be updated ~every 6 months, per WhatsApp major update.

2. Requirements

2.1. Opsec

1. A whitened WhatsApp "client" is required (SIM, TCP ports)
2. A whitened "relay" is required (UDP ports)
3. Relevant payloads should have honeypot URLs
4. SSL server for fingerprinting. This requires a difficult-to-acquire certificate. Still researching how to remove the need for this. If this is needed in production, consider having a single server in an isolated island for this purpose.
5. Throttling requirements - None currently defined.

Redacted –Export Controlled

2.3. Non-Functional Requirements

2.3.1. Packaging

1. First payload is small and fixed - not expected to change.
2. Placeholders: 1st payload is too small to contain any extra data such as Honeypot. 2nd payload will contain IS URI (including IKey) + Honeypot
3. Other exploit payloads change relatively frequently - who maintains and packages this

2.3.2. Licensing Requirements

Same restrictions as for other installations.

3. Technical Solution

3.1. New Service - WIS

Redacted –Export Controlled Redacted –Export Controlled

1. Stateless (from PS perspective) engine for mounting Heaven attack
2. WIS provides the following RESTful API to PS:
    a. Get Fingerprint (FP)
        i. Input: Target phone #, URI to whitened server, configuration (see below)
        ii. Output: Success or Failure. If Success: Target's WhatsApp version, OS, vendor, device, logs ❓
        ⚠️ Note: The solution for determining the target's WhatsApp version is currently fragile (exploits a race condition, requires a certificate...). Could be that FP will only return binary Has/Doesn't have WhatsApp installed. ⚠️
    b. Install
        i. Input: Target phone #, IKey, FP data, URI to whitened server, Payload
        ii. Output: Success or failure.
3. TBD
    Redacted –Export Controlled
    b. What placeholders are filled by PS when delivering Payload? Whitened URIs, Honeypots... ❓

JSON interface between IS and WIS as follows:

Expand source

3.2. Backend

3.2.1. PS

New configuration values - none currently required.

Keep data returned from fingerprint in relevant table ❓ As WhatsApp version becomes obsolete after max 3 weeks, consider flushing this info accordingly.

3.2.2. IS

New configuration values - none currently required.

3.3. UI Changes

1. Add Fingerprint button for Android phones. **Redacted –Export Controlled**
2. Enable zero-click if Fingerprint succeeded.

3.4. Honeypot

A separate Honeypot IP/Domain should be provided, with an IP address distinct from that used by other attacks.

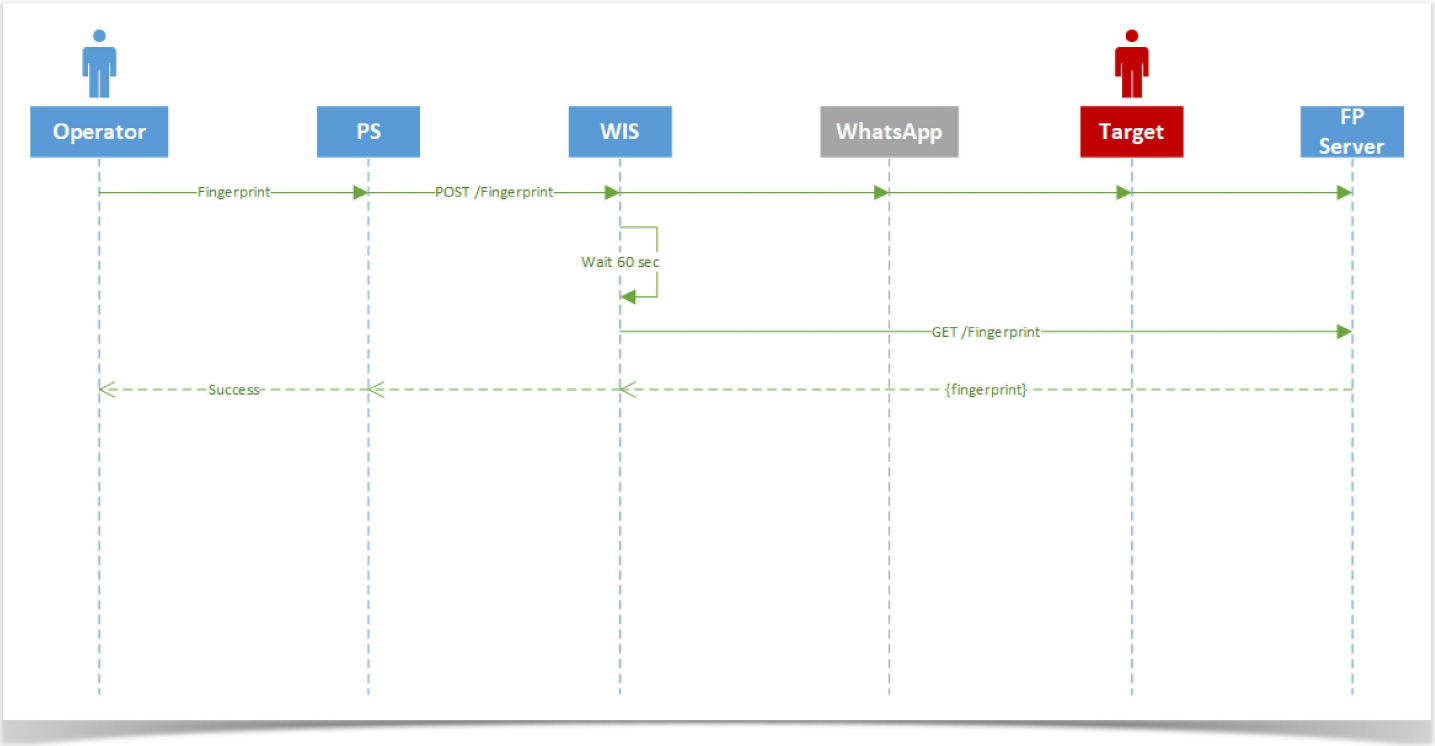
❓ Do we need separate honeypots for the relay (UDP!) and WIS ❓

4. Flows

4.1. Installation

We present two installation flows, one with a fingerprint server, in which case we get the target's WhatsApp version, and one without, in which case we only learn if the target has WhatsApp installed or not.

4.1.1. Installation with Fingerprint server



Redacted –Export Controlled

6. Emphasizes

6.1. Deployment

6.2. QA

6.3. BI

No labels

Exhibit 42

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

2.70.0.1 Technical summary

Created by [redacted] on Jun 19, 2018

This is to update that release 2.70.0.1 is approved for production.

Android executive summary:

Heaven 2.71 new capabilities:

- Support latest WhatsApp version 2.18.177 (32/64bit)

Note: All features / configurations / limitations which mentioned below for 2.70 are relevant for release 2.71 as well, pay attention that cooldown and link limitations were changed from previous release (2.70, per product guidelines)

Heaven 2.70 new capabilities:

- Support latest WhatsApp versions including 32/64 bit:
  - Support WhatsApp 2.18.122 (32bit)
  - Support WhatsApp 2.18.142 (32/64bit)
  - Support WhatsApp 2.18.156 (32/64bit)
- FFP results is now presented in the UI
  - F.g: If FFP returned results - present FFP result (Android 5.0, SM-G900F, App V2.18.79)

· Present Partial Fingerprint results in UI prior to installation, following the below matrix table:

PF #	PF Results	Message	Vectors
1	is Online + has WA + Android	Target device is Android and a candidate for Covert Android	CovertAndroid
			CraftedSMS
			OTA
2	is Offline + has WA	Target device type is unknown but there is potential for Covert Android	CovertAndroid
			CraftedSMS
			OTA
3	Is Online + has WA + Non Android	Target device is not Android and is not supported for Covert Android	CraftedSMS
			OTA

4	Non WA	Target device is not supported for covert installations	CraftedSMS
			OTA
5	Error	Fingerprint service returned an error	CraftedSMS
			OTA

- Support devices in Middle-East territory **(This option require special configuration which will be provided by demand)**
- Support Samsung S8 Oreo with kernel 4.4.13 (kernel 4.4.111 is not supported)

-

Android new capabilities:

- Support Chrome 67
- Support Samsung tablet SM-T561 3G – OS 4.4.4

Android Limitations:

- Oreo:
  - IM's:
    - No call recording
    - No support in à
      - Viber
      - Telegram
      - Telegram +
      - Signal
      - Kakao
      - WeChat
  - No support in Wiko devices via Chrome 67
  - Open bugs:
    - 76222

	Bug	Oreo > MISC > Chrome browser search history doesn't appear in the UI [SM-G950F,8.0]
• 76217	Bug	Android > Oreo > MISC > Missing details in WiFi networks [SM-G950F,8.0]

Heaven Configuration:

1.FFP should be disabled

- Disclaimer: when new WA version is released FFP should be enabled, pending Product team notification

2. Schedule attack should be enabled

3. Number of minimum accounts when **FFP disabled is 2 accounts for FP and 4 accounts for attack** (Note that optimal is to have total of 14 accounts)

4. Cooldown for attack – 2h

5. Daily link limitation: (24/cooldown)*number of attack accounts = 24/2*4 = 48

6. New configuration of release 2.70, minimum accounts available to 'install'

7. Min FP accounts : 1

8. Min Attack accounts : 1

9. The above configuration can be found in the attached PS-Settings-WIS.txt file

10 .The configuration of latest WhatsApp releases can be found in the attached SilentExploitGroupsMapping.xml, the file should be replaced with the one exist on the server

11. In PS settings.config it is required to change the Time between unsupported device failures for Heaven from “1” to “7” days as follows:

<add key="NumberOfDaysToBlockInstallOnUnsupportedAndroid" value="7" />

Platform executive summary:

- Daily Link limitation reflection to the user
  - New pop up: "Daily installation attempts limit for **Vector name** has been reached. Try again tomorrow"
  - Abuse prevention updates:

Green - currently exist

Black – 2.70 updates

Redacted – Export Controlled

Release check list:

#	Task	Approve by	Check list (V/X)
1.	QA emphasis – Link to tests	QA	**V**
1.	RN QA tips – link to tests	QA	**V**
1.	Block tests review	QA	**V**
1.	NA tests review	QA	**V**
1.	Tests status review	QA	**V**
1.	exit criteria score	QA	**V**
1.	Review all open bugs	QA+ Dev	**V**
1.	Capabilities	QA	**V**
1.	Out of scope approve by product.	Product	**V**
1.	Approval mail Is ready	Project	**V**

- **Release exit criteria score – Orange**

Thanks,

Avi

No labels

Exhibit 43

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

2.71 Android Technical summary

Created by [redacted] on Jun 20, 2018

This is to update that Android release 2.71 is approved for production.

executive summary:

Heaven 2.71 new capabilities:

- Change status code of unsupported installation due to unsupported WA version
- Fix WA crashes which was found on some devices
- Heaven fix for [redacted] clients Redacted – Per Court Order
- Support S8+, Note 8 with OS 8.X, kernel version 4.4.13 (Heaven only)
- Pre-PE flow improvement

Android Limitations:

- Oreo:
  - IM's:
    - No call recording
    - No support in à
      - Viber
      - Telegram
      - Telegram +
      - Signal
      - Kakao
      - WeChat
    - Chrome browser search history and WI-FI networks details are not available

Heaven Configuration:

1.FFP should be disabled

- Disclaimer: when new WA version is released FFP should be enabled, pending Product team notification

2.Schedule attack should be enabled

3.Number of minimum accounts when **FFP disabled is 2 accounts for FP and 4 accounts for attack** (Note that optimal is to have total of 14 accounts)

4.Cooldown for attack – 6h

5.Daily link limitation: (24/cooldown)*number of attack accounts = 24/6*4 = 16

6.New configuration of release 2.70, minimum accounts available to 'install'

7.Min FP accounts : 1

8.Min Attack accounts : 1

9.The above configuration can be found in the attached PS-Settings-WIS.txt file

10.The configuration of latest WhatsApp releases can be found in the attached SilentExploitGroupsMapping.xml, the file should be replaced with the one exist on the server

11.In PS settings.config it is required to change the Time between unsupported device failures for Heaven from “1” to “7” days as follows:

12. <add key="NumberOfDaysToBlockInstallOnUnsupportedAndroid" value="7" />

Approved versions:

Redacted – Export Controlled	
Android Package	**v2.70.230.0**
Redacted – Export Controlled	
WIS	1.0.85
UI	v2.70.1.10
Pegasus Server	2.70.8
IS Server	v2.70.4.10
Upgrade Scripts	PS Upgrade 2.64.8-2.70.8 IS Upgrade 2.64.3-2.70.4
Device parameter script	V1.0.33
Capabilities file	33953

Release check list:

#	Task	Approve by	Check list (V/X)
1.	QA emphasis – Link to tests	QA	**V**
1.	RN QA tips – link to tests	QA	**V**
1.	Block tests review	QA	**V**
1.	NA tests review	QA	**V**
1.	Tests status review	QA	**V**
1.	exit criteria score	QA	**V**
1.	Review all open bugs	QA+ Dev	**V**

1.	Capabilities	QA	**V**
1.	Out of scope approve by product.	Product	**V**
1.	Approval mail Is ready	Project	**V**

- **Release exit criteria score – Green**

Thanks,

Avi

No labels

Exhibit 44

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

Heaven accounts availability - PS behavior

Created by [redacted] last modified on Jun 06, 2018

Updated on the 06.06

Check List -

Role	Name	Date	Status
Product Manger	Moran S	02/05/18	Approved
Product Lead	Dana B		
Frontend TL	Omer		
Backend TL	Ami Y		
Solution Architect	Rony		

Motivation (Problems today)-

1. Installation can currently be initiated even when there are no WhatsApp account available (all WA accounts, FFP or attack, are either blocked by WA or expired in system)
2. Installation would fail if all WhatsApp attack accounts are in cool down (after the FFP attempt or during scheduled attack).

Requirements -

1. After pressing 'Install', check for at least 'MIN FP accounts' (should be considered if FFP is enabled) and 'MIN Attack accounts' available, i.e., not blocked or expired.
    a. If there is → continue to installation
    b. Else →
        i. UI pop up "Covert Android service is currently unavailable. Please contact your system administrator" (It was confirmed Hadas)
        ii. Add indicative log in the DB (agreed it will be in agent error log table, without ref table so it would NOT appear in the UI)
    c. Min FP accounts - X
    d. Min Attack account available - Y
    e. Support X=0 or Y=0 in the configuration (0 in configuration doesn't disable the feature, rather allows to not set a minimum number and install won't be possible to sent)

2. After FFP succeed/ FFP Failed/ FFP disabled (scheduler stage)
    a. if all attack accounts are in cool down -> continue the installation until 'installation overall time expired' (check again in X min)
        i. Agreed it will be the same mechanisms of installation expiration today (1440 sec for overall installations)
    b. if there were no accounts available during the attack stage time →installation should be failed
        i. There is no option to add indicative log, so we are assuming that Heaven installation that failed due to expiration limit will be because the attack accounts were unavailable.

No labels

5 Comments

Unknown User (omers)

@[redacted] Any updates regarding the error message you want to display? What are the options under the XXX?

[redacted]

I've just sent the option above to Hadas, I will let you know until tomorrow the final message

[redacted]

see message above, thanks!

Unknown User (omers)

@Unknown User (amiy) Is there a separation between FP accounts and attack accounts? seems like there is only one "NoValidAccountsCovertAndroid" error.

Unknown User (iditb)

When MinimumWhatsAppFingerprintAccountsRequired / MinimumWhatsAppAttackAccountsRequired = 0, and there is 0 available accounts, PS **doesn't allow** to send installation.

Meaning, 0 in configuration doesn't disable the feature like the PRD requires, rather allows to not set a minimum number.

[redacted] approved this behavior, since it doesn't make any sense to allow sending FFP/installation when PS knows no accounts are available.

Exhibit 45

UNREDACTED VERSION OF DOCUMENT PROPOSED

TO BE FILED UNDER SEAL

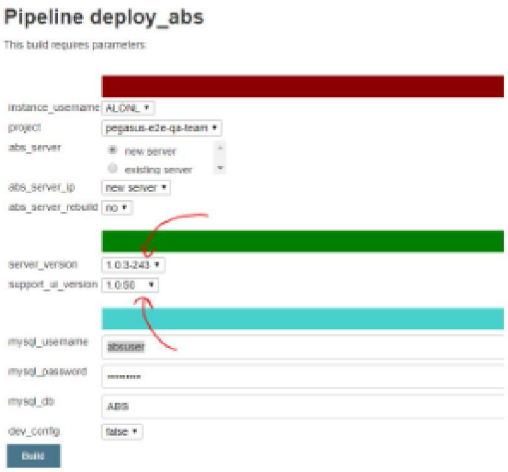
How to create ABS server

Created by [redacted] last modified on May 01, 2019

ABS server replace the WIS server for Eden, you can use ABS server IP Redacted – Per Court Order or create new one.

In order to create a new ABS server the following steps are need to perform:

1. Login to https://jenkins.nsogroup.com:8443/view/Android/ and navigate to deploy_abs job.
2. Choose the wanted ABS server version and supported UI



3. Validate that SQL set with user: absuser and password: Aa1234567890

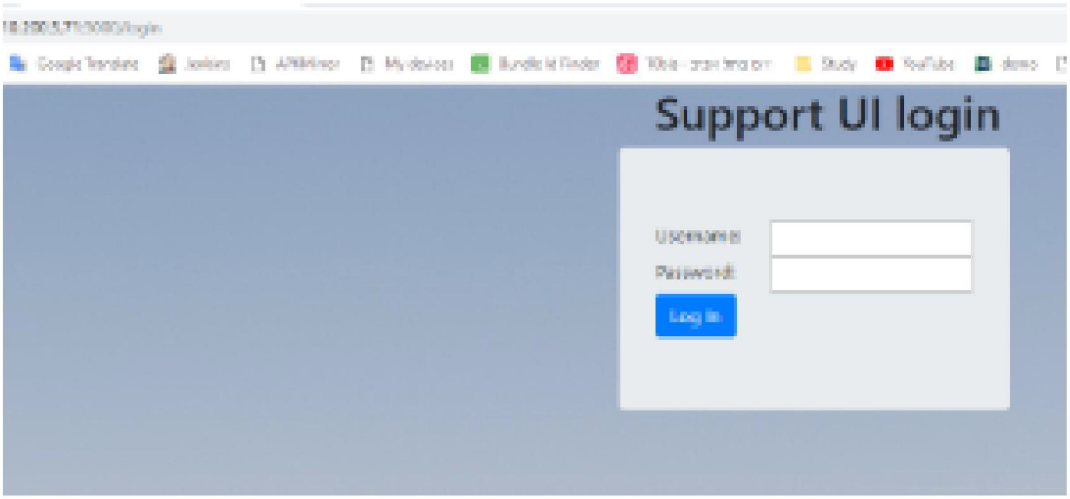
4. Once the job is done open Putty (or MRemote) and connect to the server, the IP should be in Jenkins email, with **Username**: qadmin , **Password**: 5Hhnyt9d and grant yourself **sudo** permissions by running the following command **sudo su**

5. In order to configure config.json you can:

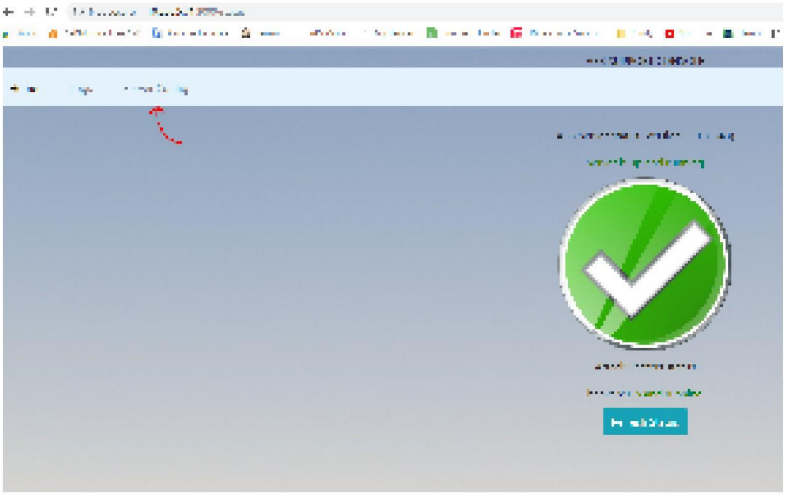
* Run the following command **nano /opt/abs/server/attackVectors/heaven/config.json**

OR

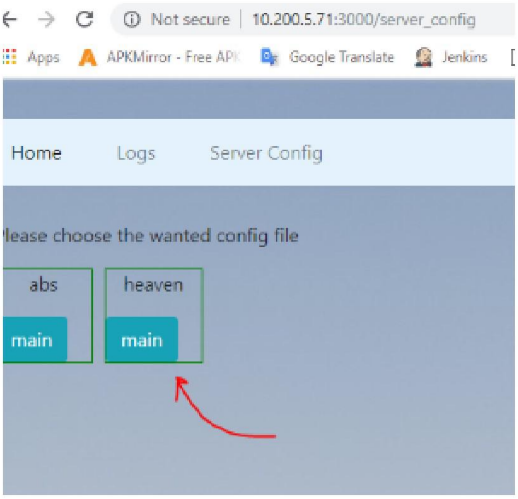
* Use the UI by connecting in your browser to http://<ABS Server IP>:3000/login and login using **Username**: qadmin , **Password**: 5Hhnyt9d



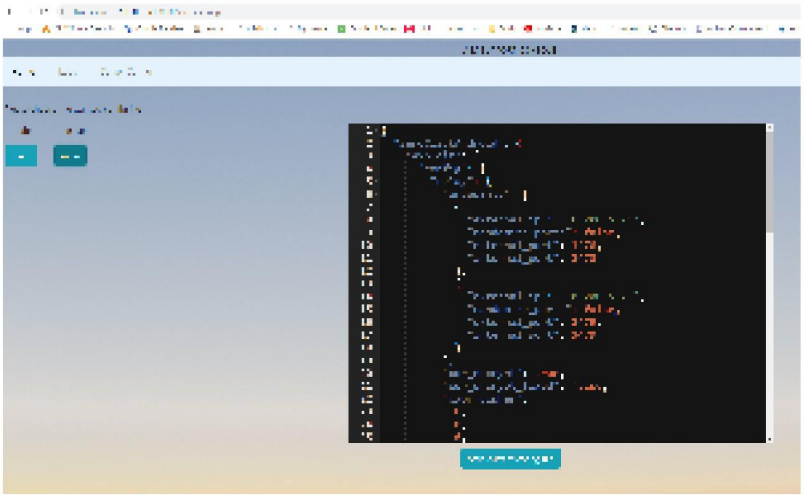
Click on Server Config



Click on hevan



Config.json file can be updated from here and save, but the systemctl restart ABS still must be done in the console, not supported from the UI



All **'relay', 'stager' and 'microd'** to endpoints localhost values should be replaced with ABS IP. **'signaling'** IP should hold WA Server IP

6. Make sure to save the file and run **systemctl restart abs** to restart the server process to apply all the changes. And run the following command **systemctl status abs** to verify the restart was performed and the service is up and running

Now your ABS server is ready.

No labels