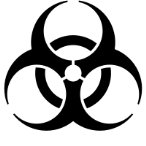


****

United Healthcare Group – Cyber Defense

Optum Advanced Field Examiner (OAFE) – Data Sheet

DRAFT

**Version 1.0**

**Version History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version Number** | **Implemented By** | **Revision Date** | **Approved By** | **Approval Date** | **Reason** |
| 0.1 | Rich Baker | 7/22/2016 |  |  | Initial Document Draft |
| 0.9 | Alyn, Inc. | 7/27/2016 |  |  | Complete Document Update |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Table of Contents**

[1. OAFE Overview 4](#_Toc457479141)

[2. OAFE Security 4](#_Toc457479142)

[3. OAFE Hardware 4](#_Toc457479143)

[4. OAFE Capabilities 4](#_Toc457479144)

[4.1. Network Malware Inspection 4](#_Toc457479145)

[4.2. Data Analytics and Visualization 5](#_Toc457479146)

[4.3. Endpoint Forensic Investigation 5](#_Toc457479147)

[4.4. Incident Response Ticketing 5](#_Toc457479148)

[4.5. Log Management 6](#_Toc457479149)

[4.6. File Analysis 6](#_Toc457479150)

[4.7. Handle Network Interactions 7](#_Toc457479151)

[4.8. Examine Memory Snapshots 8](#_Toc457479152)

[5. UHG/Optum Technology System to AOFE Tool Mapping 9](#_Toc457479153)

[6. OAFE Application Ports 9](#_Toc457479154)

# 

OAFE Mascot

# OAFE Overview

The Optum Advanced Field Examiner (OAFE) is a custom built server providing a central analytical and forensic toolkit for security investigations. The OAFE is a purpose built incident response and forensics system, designed for rapid deployment in remote environments, and comes pre-loaded with open source tools providing essential capabilities necessary to effectively analyze end points, logs, and network activity for security anomalies and malicious activity and stage incident response actions. While maintaining a robust set of capabilities, the OAFE is not an active network sensor or fulltime collection system. The OAFE collects information only at a point in time, after the identification of a possible security threat and being activated by an Optum Cyber Defense (CD) analyst.

# OAFE Hardware

The OAFE’s current data center requirements are 2U of rack space, 2 power (NEMA) connections, and 4 Gigabit network connections. The OAFE servers are currently (as of July 2016) configured on a 2U Hewlett Packard ProLiant DL380 Gen 9 chassis with dual XEON processors and 128GB of memory running Ubuntu 16.04 LTS. Storage capabilities are custom configured per client based on the number of end points. OAFE can be installed and ran on a system with reduced hardware components as described above, if necessary, but performance will be significantly impacted.

# OAFE Security

The OAFE is built with security in mind, from analysis of security events and activity to the configuration of the system itself. The underlying software is fully patched and best business practice security controls enabled. Further access to the system requires multi-factor authentication, analyst accounts are issued only to personnel assigned to conduct the investigation, and connections to the OAFE servers are only allowed from the Optum CD CFI network. Additionally, all data collected by the OAFE server stays resident on the server, and the hard drives within the OAFE are configured to be protected via full disk encryption by default.

# OAFE Capabilities

The OAFE provides an environment where Analysts have immediate access to various tools necessary to conduct endpoint, forensic, file, log and network analysis within the client network. In addition to the applications listed below, the OAFE runs Oracle VirtualBox, which allows for a custom virtual machine to be built matching the client’s local endpoint specifications in order to join the client’s network for additional incident response and scanning activity. Further, this virtualization capability allows CD Analysts to utilize custom analytical environments, such as SIFT (SANS Investigative Forensics Toolkit) which contains robust Forensic Analysis tools.

## Endpoint Forensic Investigation

The OAFE endpoint forensic investigation applications enable the analysts to proactively analyze network endpoints such as workstations and servers, which are detected exhibiting anomalous activity or have been identified as potentially infected or compromised.

* Commercial (require additional license): EnCase Enterprise, F-Response – EnCase Enterprise and F-Response are both commercial tools which require additional licensing to utilize. Both enable an analyst to remotely collect and analyze system data in a forensically sound manner. EnCase Enterprise contains additional analytical capabilities reducing the time needed to conduct analysis of suspect systems.
* Open Source: Google Rapid Response, Kansa – Google Rapid Response (GRR) is an agent based endpoint analysis tool which allows investigators to remotely acquire system and file data, sweep the enterprise for identified IOCs, and analyze suspected infected or compromised Windows, Linux and OSX based systems. Kansa utilizes PowerShell code to collect system information from Windows based systems. Due to PowerShell being installed by default in most current Windows deployment, no agent is required. Kansa can sweep a Windows Environment for specific indicators very rapidly.

## Examine Memory Snapshots

As malware runs, portions of the code are resident in the systems memory. OAFE contains tools to enable analysts to analyze memory dumps acquired during the course of an investigation. More sophisticated malware variants run almost exclusively within memory, making memory analysis even more necessary.

* [Volatility Framework](https://github.com/volatilityfoundation/volatility)
* [findaes](http://jessekornblum.livejournal.com/269749.html), AESKeyFinder, RSAKeyFinder,
* [VolDiff](https://github.com/REMnux/docs/blob/master/tools/VolDiff.md)
* [Rekall](http://www.rekall-forensic.com/)
* [linux\_mem\_diff\_tool](https://github.com/monnappa22/linux_mem_diff_tool)

## File Analysis

During Incident Response and forensics investigations, the analysts will discover suspicious and openly malicious files. OAFE provides a safe environment for these files to be quickly analyzed and additional IOCs identified for further analysis within the enterprise. The following tools provide analysts with various capabilities required to effectively analyze these files.

#### Browser Malware

* Website analysis: [Thug](https://github.com/buffer/thug), [mitmproxy](http://mitmproxy.org/), [Network Miner Free Edition](http://www.netresec.com/?page=NetworkMiner), [curl](http://curl.haxx.se/), [Wget](https://www.gnu.org/software/wget/), [Burp Proxy Free Edition](http://portswigger.net/burp/), [Automater](http://www.tekdefense.com/automater/), [pdnstool](https://github.com/chrislee35/passivedns-client), [Tor](https://www.torproject.org/), [tcpextract](http://tcpxtract.sourceforge.net/), [tcpflow](https://github.com/simsong/tcpflow), [passive.py](https://github.com/REMnux/distro/blob/v6/passive.py), [CapTipper](https://github.com/omriher/CapTipper), [yaraPcap.py](https://github.com/kevthehermit/YaraPcap)
* Flash: [xxxswf](http://hooked-on-mnemonics.blogspot.com/2011/12/xxxswfpy.html), [SWF Tools](http://www.swftools.org/), [RABCDAsm](https://github.com/CyberShadow/RABCDAsm), [extract\_swf](https://gist.github.com/noonat/821548), [Flare](http://www.nowrap.de/flare.html)
* Java: [Java Cache IDX Parser](https://github.com/Rurik/Java_IDX_Parser/), [JD-GUI Java Decompiler](http://jd.benow.ca/), [JAD Java Decompiler](http://varaneckas.com/jad), [Javassist](http://www.javassist.org/), [CFR](http://www.benf.org/other/cfr/)
* JavaScript: [Rhino Debugger](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Rhino/Debugger), [ExtractScripts](http://blog.didierstevens.com/programs/extractscripts/), [SpiderMonkey](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey), [V8](https://code.google.com/p/v8/), [JS Beautifier](https://github.com/einars/js-beautify)

#### Document Files

* PDF: [AnalyzePDF](https://github.com/hiddenillusion/AnalyzePDF), [Pdfobjflow](http://www.aldeid.com/wiki/Pdfobjflow), [pdfid](http://blog.didierstevens.com/programs/pdf-tools/), [pdf-parser](http://blog.didierstevens.com/programs/pdf-tools/), [peepdf](http://eternal-todo.com/tools/peepdf-pdf-analysis-tool#releases), [Origami](https://code.google.com/p/origami-pdf/), [PDF X-RAY Lite](https://github.com/9b/pdfxray_lite), [PDFtk](http://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/), [swf\_mastah](http://blog.9bplus.com/snatching-swf-from-pdfs-made-easier/), [qpdf](http://qpdf.sourceforge.net/), [pdfresurrect](https://github.com/enferex/pdfresurrect)
* Microsoft Office: officeparser, pyOLEScanner.py, oletools, libolecf, oledump, emldump,MSGConvert, base64dump.py, unicode
* Shellcode: [sctest](http://libemu.carnivore.it/), unicode2hex-escaped, unicode2raw, [dism-this](http://hooked-on-mnemonics.blogspot.com/2012/10/dism-thispy.html), [shellcode2exe](https://github.com/MarioVilas/shellcode_tools/blob/master/shellcode2exe.py)

#### Extract and Decode Artifacts

* Deobfuscate: [unXOR](https://github.com/tomchop/unxor/), [XORStrings](http://blog.didierstevens.com/2013/04/15/new-tool-xorstrings/), [ex\_pe\_xor](http://hooked-on-mnemonics.blogspot.com/2014/04/expexorpy.html), [XORSearch](http://blog.didierstevens.com/programs/xorsearch/), [brxor.py](https://github.com/REMnux/distro/blob/v6/brxor.py), [xortool](https://github.com/hellman/xortool), [NoMoreXOR](https://github.com/hiddenillusion/NoMoreXOR), [XORBruteForcer](http://eternal-todo.com/category/bruteforce), [Balbuzard](https://bitbucket.org/decalage/balbuzard/wiki/Home), [FLOSS](https://github.com/fireeye/flare-floss/)
* Extract strings: [strdeobj](http://totalhash.com/download/strdeob.pl.txt), [pestr](http://pev.sourceforge.net/), [strings](http://en.wikipedia.org/wiki/Strings_(Unix))
* Carving: [Foremost](http://foremost.sourceforge.net/), [Scalpel](http://www.forensicswiki.org/wiki/Scalpel), [bulk\_extractor](http://www.forensicswiki.org/wiki/Bulk_extractor), [Hachoir](https://bitbucket.org/haypo/hachoir)

#### Process Multiple Malware Samples

* [Maltrieve](https://github.com/technoskald/maltrieve), [Ragpicker](https://code.google.com/p/malware-crawler/), [Viper](https://github.com/botherder/viper), [MASTIFF](https://git.korelogic.com/mastiff.git/), [Density Scout](http://www.cert.at/downloads/software/densityscout_en.html)

#### Examine File Properties and Contents

* Define signatures: [YaraGenerator](https://github.com/Xen0ph0n/YaraGenerator), [IOCextractor](https://github.com/stephenbrannon/IOCextractor), [Autorule](http://joxeankoret.com/blog/2012/04/29/extracting-binary-patterns-in-malware-sets-and-generating-yara-rules/), [Rule Editor](https://github.com/ifontarensky/RuleEditor), [ioc-parser](https://github.com/armbues/ioc_parser)
* Scan: [Yara](http://plusvic.github.io/yara/), [ClamAV](http://www.clamav.net/), [TrID](http://mark0.net/soft-trid-e.html), [ExifTool](http://www.sno.phy.queensu.ca/~phil/exiftool/), [virustotal-submit](http://blog.didierstevens.com/programs/virustotal-tools/), [Disitool](http://blog.didierstevens.com/programs/disitool/)
* Hashes: [nsrllookup](https://github.com/rjhansen/nsrllookup), [Automater](http://www.tekdefense.com/automater/), [Hash Identifier](https://code.google.com/p/hash-identifier/), [totalhash](https://gist.github.com/malc0de/10270150), [ssdeep](http://ssdeep.sourceforge.net/), [virustotal-search](http://blog.didierstevens.com/programs/virustotal-tools/),[VirusTotalApi](https://github.com/doomedraven/VirusTotalApi)

#### Investigate Linux Malware

* System: [Sysdig](http://www.sysdig.org/), [Unhide](http://www.unhide-forensics.info/)
* Disassemble: [Vivisect](http://visi.kenshoto.com/viki/Vivisect), [Udis86](http://udis86.sourceforge.net/), [objdump](http://en.wikipedia.org/wiki/Objdump)
* Debug: [Evan’s Debugger (EDB)](http://codef00.com/projects#debugger), [GNU Project Debugger (GDB)](http://www.sourceware.org/gdb/)
* Trace: [strace](https://sourceforge.net/projects/strace/), [ltrace](http://ltrace.org/)
* Investigate: [Radare 2](https://github.com/radare/radare2), [Pyew](https://code.google.com/p/pyew/), [Bokken](https://inguma.eu/projects/bokken), [m2elf](https://github.com/XlogicX/m2elf), [ELF Parser](http://elfparser.com/)
* Automated Dynamic Analysis: Cuckoo Sandbox

#### Edit and View Files

* Text: [SciTE](http://www.scintilla.org/SciTE.html), [Geany](http://www.geany.org/), [Vim](http://www.vim.org/)
* Images: [feh](http://feh.finalrewind.org/), [ImageMagick](http://www.imagemagick.org/)
* Binary: [wxHexEditor](http://sourceforge.net/projects/wxhexeditor/), [VBinDiff](http://www.cjmweb.net/vbindiff/)
* Documents: [Xpdf](http://www.foolabs.com/xpdf/)

#### Statically Examine PE Files

* Unpacking: [UPX](http://upx.sourceforge.net/), [Bytehist](https://www.cert.at/downloads/software/bytehist_en.html), [Density Scout](http://www.cert.at/downloads/software/densityscout_en.html), [PackerID](http://handlers.sans.org/jclausing/packerid.py)
* Disassemble: [objdump](http://en.wikipedia.org/wiki/Objdump), [Udis86](http://udis86.sourceforge.net/), [Vivisect](http://visi.kenshoto.com/viki/Vivisect)
* Find anomalies: [Signsrch](http://aluigi.altervista.org/mytoolz.htm), [pescanner](https://code.google.com/p/malwarecookbook/source/browse/trunk/3/8/pescanner.py), [ExeScan](http://securityxploded.com/exe-scan.php), [pev](http://pev.sourceforge.net/), [Peframe](https://github.com/guelfoweb/peframe), [pedump](http://pedump.me/)
* Investigate: [Bokken](https://inguma.eu/projects/bokken), [RATDecoders](https://github.com/kevthehermit/RATDecoders), [Pyew](https://code.google.com/p/pyew/), [readpe.py](https://github.com/crackinglandia/pype32), [PyInstaller Extractor](https://github.com/zrax/pycdc), [DC3-MWCP](https://github.com/Defense-Cyber-Crime-Center/DC3-MWCP)
* Automated: [Cuckoo Sandbox](https://github.com/cuckoosandbox/cuckoo), [Viper Framework](http://viper.li/)

#### Investigate Mobile Malware

[Androwarn](https://github.com/maaaaz/androwarn), [AndroGuard](https://github.com/androguard/androguard), [Cuckoo Sandbox](https://github.com/cuckoosandbox/cuckoo)

## Network Malware Inspection

Network malware inspection allows OAFE to operate as a passive network and file inspection appliance, monitoring for known malicious traits within network traffic attributes and file attributes being transferred over the network.

* Maltrail - Maltrail is a malicious traffic detection system, utilizing publicly available and custom lists of malicious indicators of compromise (IOCs) for both network traffic (domains, IP addresses and User Agents) as well as malicious files. Maltrail monitors network traffic passing through the OAFE for matching network or file based indicators from the IOC lists ingested.

## Handle Network Interactions

In addition to analyzing files for indicators that can determine capability and intent of the attacker as well as indicators of additional activity within the network, the OAFE contains tools for dynamic malware analysis and analyzing network traffic. All malware eventually communicates with an external system, the tools and utilities within OAFE allow analysts to investigate that communication or intercept/mimic the traffic in the analysis environment.

* [Wireshark](http://www.wireshark.org/), [ngrep](http://ngrep.sourceforge.net/), [TCPDump](http://www.tcpdump.org/), [tcpick](http://tcpick.sourceforge.net/)
* Services: [FakeDNS](http://code.activestate.com/recipes/491264-mini-fake-dns-server/), [Nginx](http://nginx.org/), [fakeMail](http://sourceforge.net/projects/fakemail/), [Honeyd](http://www.honeyd.org/), [INetSim](http://www.inetsim.org/), [Inspire IRCd](http://www.inspircd.org/), [OpenSSH](http://www.openssh.com/), accept-all-ips, passivedns monitoring
* Miscellaneous network: [prettyping.sh](https://bitbucket.org/denilsonsa/small_scripts/src/3ec16014c839ea0852fae492813ad2293bd61155/prettyping.sh), set-static-ip, renew-dhcp, [Netcat](http://netcat.sourceforge.net/), [EPIC IRC Client](http://www.epicsol.org/),[stunnel](https://www.stunnel.org/), [Just-Metadata](https://github.com/ChrisTruncer/Just-Metadata)
* Deep Packet Inspection: Moloch
* Netflow: FlowBAT, NTOPng, SiLK

## Data Analytics and Visualization

Data analytics and visualization enable the OAFE to operate as a micro-instance of a Big Data platform, giving analysts the ability to collect and parse massive amounts of network, security and device logs more effectively and efficiently. The Visualization capabilities enable analysts to rapidly identify outliers in the data sets and present the information in an easy to digest and present format.

* Search: ElasticSearch – ElasticSearch enables unstructured data to be queried and analyzed very rapidly without the need to define a data schema first.
* Log Ingest: Beats, Logstash – Beats and Logstash consumes data sources of all types and variations and ships it to ElasticSearch for further searching and analysis.
* Visualization: Kibana, Kibi – Kibana and Kibi provide a visualization platform and real-time summary and charting for data being ingested by Beats and Logstash or analyzed by ElasticSearch.

## Log Management

* GrayLog, syslog – Graylog and Syslog both facilitate the collection of logs from various log sources providing incident investigators the broadest view of data by obtaining information from multiple log sources for incident correlation and evaluation.

## Incident Response Ticketing

As suspicious activity is identified and analyzed, OAFE provides a platform for tracking and documenting all analytical activity.

* Fast Incident Response (F.I.R.) – FIR is a cybersecurity incident management platform designed with agility and speed in mind. It allows for easy creation, tracking, and reporting of cybersecurity incidents. Proper tracking and incident documentation is absolutely necessary for effective incident response and FIR provides the structure for this documentation.

# UHG/Optum Technology System to AOFE Tool Mapping

Each of the tools and applications within the OAFE directly map to a similar commercial enterprise level tool utilized within the UHG environment.

|  |  |  |  |
| --- | --- | --- | --- |
| Technology | UHG Internal System | OAFE Tools | Notes |
| Deep Packet Inspection | RSA Security Analytics | Moloch |  |
| Network Malware Inspection | FireEye | Maltrail |  |
| Endpoint Forensics | EnCase Enterprise, F-Response | Google Rapid Response, EnCase Enterprise, F-Response | EnCase and F-Response require valid licenses |
| Incident Response Ticketing | RiskVision | Fast Incident Response | This requires consulting time at an additional fee. |
| Data  Analytics and Visualization | Security Big Data Lake, Elasticsearch, Logstash, Kibana | Elasticsearch, Logstash, and Kibana | Additional consulting fees are required for configurations that involve anything more than basic search functionality for CFI Investigators. |
| Log Management | HP ArcSight, Security Big Data Lake | Graylog (ELK Stack required) | This is enabled on a fee basis. Monitoring is an additional charge. |
| Malware – Dynamic Analysis | ThreatAnalyzer, Cuckoo Sandbox | Cuckoo Sandbox |  |

# OAFE Application Ports

|  |  |  |  |
| --- | --- | --- | --- |
| Application | Local access (loopback) | Internal Access | Notes |
| Cuckoo Sandbox Web Interface | 8511 (tcp) | 8501 (tcp) | Proxied via nginx. Other languages can be configured, if required. |
| Cuckoo Sandbox API | 8510 (tcp) | 8504 (tcp) |  |
| Maltrail | 8338 (tcp) | 8338 (tcp) | Internal authentication provided by Maltrail |
| Google Rapid Response | 8000 (tcp) | 8000 (tcp), 8080 (tcp – agent access) | Agents access the server on port 8080. Port 8000 is management interface. |
| Kibana | 5601 (tcp) | 8502 (tcp) | Proxied via nginx |
| Elasticsearch | 9200 (tcp) | N/A | Can be configured through proxy. |
| NTOPng | 3000 (tcp) | 3000 (tcp) | Internal authentication |
| Graylog | 9000 (tcp) | 9000 (tcp), 12900 (tcp) | Web console is accessed via port 9000 |
| Viper | 8514 (tcp) | 8505 (tcp) |  |
| Kibi | 5606 (tcp) | 8507 (tcp) |  |
| Moloch (DPI) | 8005 (tcp) | 8005 (tcp) | Internal authentication |
| Logstash | 5004 (tcp) | N/A |  |