8484 Georgia Avenue
Suite 940
Silver Spring, MD 20910
Phone: 301.641.8252
Fax: 301.576.4590

Jeff Janies
Lead Analyst
Jeff.janies@redjack.com
jeffery.l.janies.ctr@mail.mil

Paul Avellino
Program Manager
paul.avellino@redjack.com
paul.d.avellino.ctr@mail.mil

Data Generation Specifications

Version 1

Version	Date	Author	Revisions
1	August 4, 2015	Jeff Janies	Initial Version

References:

None

1 Introduction

This document outlines the output types and behaviors generated by the data generation script requested by ID6. The purpose of this script is to produce realistic data samples in representative formats that can be used by the DoD community for analytic development. The script tightly defines a set of common activities normally found on the DoDIN, and defines how they would be represented in four distinct output formats: SiLK, Sourcefire Common Event Format (CEF), Web Content Filter (WCF), and HBSS CEF.

Sufficiently diverse and accurate data samples for testing are a common challenge for analytic development within the DoD community. Commonly, analysts must rely on assumptions about the data and its format, which inevitably results in additional integration effort when the assumptions are proven wrong.

The remainder of the paper is structured as follows: Section 2 defines the various output types produced by the data generation script and Section 3 defines the various activities represented within.

2 Output Types

The data generation tool provides outputs four common formats used by DoD analysts, which are shown the following table:

Name	Format	File Name
Silk	"|"-delimited Text	silk.txt
Sourcefire	CEF	source_fire.txt
WCF	CSV	wcf.txt
HBSS	CEF	hbss.txt

2.1 SiLK

The data generation script simulates the SiLK data that would be produced by the IAP sensors at the DoDIN border and available in the ISR class in CENTAUR. As with the source it simulates, the data generation script outputs unidirectional network traffic flow records in which each record summarizes communications between sources and destinations on a network as a set of packets related closely in time. Unlike SiLK as stored in CENTAUR, the SiLK data provided is in text from instead of binary SiLK. The `rwtuc` command can be used to reproduce the binary form stored in CENTAUR.

For the purposes of this version of the data generation script the *initialFlags*, *sessionFlags*, and *attributes* fields are blank.

Output Format

The fields for the SiLK output will appear in the following order, separated by pipe delimiters. This header will always be the first line of the *silk.txt* output file:

```
sIP|dIP|sPort|dPort|protocol|packets|bytes|flags|sTime|dur|eTime|sensor|
sType|dType|scc|dcc|class|type|iType|iCode|icmpTypeCode|in|out|nhIP|
initialFlags|sessionFlags|attributes|application
```

Example Record

The following record represents one line of a silk.txt output file. Note that fields for which there is no data are left as empty strings.

```
192.168.135.16|4.118.79.214|38397|53|17|1|238||2015/07/31T14:02:49|0|
2015/07/31T14:02:49|0|||United States|United States|isr|in||||||0.0.0.0||||
```

2.2 Sourcefire alerts

The date generation script simulates Sourcefire deployed IAP. The script generates a Sourcefire alert network traffic matches a pre-defined signature, currently this will be either a "`Malicious code download detected`" or "`IRC traffic detected`".

Output Format

Each Sourcefire record begins with the following header:

```
CEF:0|ArcSight|ArcSight|7.0|agent:000|Agent [Sourcefire_D] type |LOW|
```

After the header, the record has an extension field in CEF format, which contains a set of key-value pairs. A single space separates each key-value pair. For more information about the key-value pairs, see the ArcSight Common Event Format Guide[1].

[1] https://protect724.hp.com/docs/DOC-1072

Example Record

This record represents one line of a sourcefire.txt output file.

```
CEF:0|ArcSight|ArcSight|7.0|agent:000|Agent [Sourcefire_D] type |LOW|
eventId=0 externalId=6 start=1438351373531 end=0 proto=TCP catdt=Network-
based IDS/IPS art=1432046499771 cat=unknown deviceSeverity=Low act=gray --
unknown rt=0 devicePayloadId=106|66889 src=192.168.243.38 sourceZoneURI=/All
Zones/ArcSight System/Public spt=63283 dst=214.114.77.233
destinationZoneURI=/All Zones/ArcSight System/Public dpt=80 cs1=fdjksla;
cs2= cs3= cs4= cs5=CVE_2015_0000 cs6=Malicious code download detected cn1=
cn2= cn3= cs1Label=payload cs2Label=Fingerprint cs3Label=HostType
cs4Label=ClientApplicationname cs5Label=CVEId cs6Label=SnortId
cn1Label=BugtraqId cn2Label=FingerpritId cn3Label=BlockType
deviceCustomDate1Label=LastUsed deviceCustomDate2Label=HostLastSeen
c6a2Label=Source IPv6 Address c6aLabel=Destination IPv6 Address
ahost=client.hostname agt=9.9.9.9 agentZoneURI=/All Zones/Arcsight
System/Public Address Space Zones av=7.0 atz=Zulu aid=2134-1=zADC\=\=
at=sourcefire_api dvc=2.2.2.2 deviceZoneURI=/All Zones/ArcSight
System/Public Address Space Zones dtz=Zulu deviceExternalId=106_cefVer=0.1
ad.RecordType.i=7 ad.RecordLength.i=60 ad.DetectionEnginUuid.i=12345ABD
ad.DetectionEngineDescription=ABD-BACD ab
ad.EventMicrosecond.d=1438351373531 ad.DetectionEngineName=I_AM_AN_ENGINE
ad.RuleRevisionUuid.i=1234678ABCD ad.RuleRevision.i=1
ad.RuleUuid.i=12345678ABDCD
```

2.3 Web Content Filter (WCF) Records

The data generation script simulates the activity from the Web Content Filter (WCF). WCF is a web proxy that filters and logs web requests made from DoDIN hosts within the network. Some of the traffic is allowed and recorded; other traffic perceived as malicious is blocked and labeled as a threat.

Output Format

A WCF record contains a series of fields, with fields separated by commas. There are two types of WCF records: TRAFFIC records, and THREAT records[2].

The fields for a TRAFFIC record appear in the following order:

```
"FIRST", "Receive Time", "Serial Number", "Type", "Subtype", "FIRST",
"OTHER_DATETIME", "Source IP", "Destination IP", "NAT Source IP", "NAT
Destination IP", "Rule Name", "Source User", "Destination User",
"Application", "Virtual System", "Source Zone", "Destination Zone", "Ingress
Interface", "Egress Interface", "Log Forwarding Profile", "OTHER_DATETIME",
"Session ID", "Repeat Count", "Source Port", "Destination Port", "NAT Source
Port", "NAT Destination Port", "Flags", "Protocol", "Action", "some_num1",
"some_num2", "some_num3", "some_num4", "OTHER_DATETIME", "small_num",
"Category", "ZERO", "scc", "dcc"
```

The fields for a THREAT record appear in the following order:

[2] Note, headers were not provided with the WCF data. Therefore, some of the field's headers are ambiguous.

```
FIRST, Receive Time, Serial Number, Type, Subtype, FIRST, OTHER_DATETIME,
Source IP, Destination IP, NAT Source IP, NAT Destination IP, Rule Name,
Source User, Destination User, Application, Virtual System, Source Zone,
Destination Zone, Ingress Interface, Egress Interface, Log Forwarding
Profile,
OTHER_DATETIME, Session ID, Repeat Count, Source Port, Destination Port,
NAT Source Port, NAT Destination Port, Flags, Protocol, Action,
Miscellaneous, Threat ID, Category, Severity, Direction, scc, dcc
```

Example Record

The following represents one TRAFFIC record from a wcf.txt output file:

```
1,07/31/2015 14:02:50,001901000236,TRAFFIC,url,1,07/31/2015
14:02:50,192.168.135.16,202.236.70.107,192.168.1.1,192.168.1.1,WCF_GEOIP_USN
_Allow_out,,,web-
browsing,vsys1,Niprnet,Internet,ethernet1/22,ethernet1/21,Splunklog-
Universal,07/31/2015
14:02:50,10,0,38410,80,0,0,0x8000,tcp,allow,198,1150,198,1,07/31/2015
14:02:50,56,unknown,0,United States,United States
```

The following represents one THREAT record from a wcf.txt output file:

```
1,07/31/2015 14:11:36,001901000236,THREAT,url,1,07/31/2015
14:11:36,192.168.227.158,20.187.242.102,192.168.1.1,192.168.1.1,IOC-
Filter,,,web-
browsing,vsys1,Niprnet,Internet,ethernet1/22,ethernet1/21,Splunklog-
Universal,07/31/2015 14:11:36,10,0,65269,80,0,0,0x8000,tcp,block-
url,http://www.baddomain.com/downloadvirus.exe,(9999),unknown,informational,
client_to_server,United States,China
```

2.4 HBSS Event Data

The data generation script simulates Host-Based Security System (HBSS) instances running on DoDIN devices. HBSS is a host-based intrusion prevention system that monitors a single host for suspicious activities. After identifying a malicious activity, HBSS logs information about the activity, attempts to block or stop it, and then reports it.

Output Format

Each HBSS record begins with the following header:

```
CEF:0|McAfee|ePolicy Orchestrator|4.6.8|1095|Access Protection rule
violation detected and NOT blocked|Low|
```

After the header, the record has an extension field in CEF format, which contains a set of key-value pairs. A single space separates each key-value pair. For more

information about the key-value pairs, see the ArcSight Common Event Format Guide[3].

Example Record

This record represents one line of an hbss.txt output file.

```
CEF:0|McAfee|ePolicy Orchestrator|4.6.8|1095|Access Protection rule
violation detected and NOT blocked|Low| eventId=0 externalId=0 msg=Threat
categorySignificance=/Informational/Alert categoryBehavior=/Access
categoCEFryTechnique=/Policy/Breach categoryDeviceGroup=Firewall
catdt=Network-based categoryOutcome=/Success
categoryObject=/Host/Application/Service art=1438351373531 cat=0
deviceSeverity=5 act=Permitted rt=1438351373531 src=214.114.77.233
sourceZoneURI=/All Zones/ArcSight System/Private Address Space
Zones/RFC1918: 0.0.0.0 - 255.255.255.255 sproc=0 dhost=SMDCDC05 dst=0.0.0.0
destinationZoneURI=/All Zones/ArcSight System/Private AddressSpace
Zones/RFC1918: 0.0.0.0-255.255.255.255 dpt=0 duser=N/A
fname=downloadvirus.exe filePath=~/downloadvirus.exe cs1=Malicious code
download detected cs2=Malware cs3=VirusScan Enterprise cs4=8.8 cs5=330D
flexString1=SMDCDC05 deviceCustomDate1=1438351373531
deviceCustomDate2=1438351373531 cs1Label=VirusName cs2Label=VirusType
cs3Label=ProductName cs4Label=ProductVersion cs5Label=Agent GUID
cs6Label=DATVersion deviceCustomDate1Label=Generated Time (UTC)
deviceCustomDate2Label=DetectTime ahost=client.hostname agt=192.168.27.87
agentZoneURI=/All Zones/ArcSight System/Public Address Space Zones/
av=7.0.1.6963.0 atz=Zulu aid=3aGJDCkkBABCAAejGq56ZpQ\=\= at=syslog
dvchost=host dvc=0.0.0.0 deviceZoneURI=/All Zones/ArcSight System/Public
Address Space Zones/ dtz=UTZ deviceFacility=NIPRNET _cefVer=0.1
ad.dvc.4=0.0.0.0 ad.EVTID=01 ad.EVTTIME.d=1438351373531 ad.dtz=Etc/GMT
ad.DETECTINGPRODUCTIPV6=FSDDDDD ad.dvchost=hostname ad.art=1438351373531
ad.SIGNATUREID.l=0 ad.rt=1438351373531 ad.DETECTINGPRODUCTIPV4.l=00000000
ad.sourceZoneID=MCtzU4ycB ad.destinationZoneID=MCtzU4ycB
eventAnnotationVersion=1 ad.customerURI=All Customers/RCERT-C
eventAnnotationManagerReceiptTime=1438351373531 severity=0 relevance=10
priority=5 eventAnnotationStageURI=/All Stages/Queued
ad.arcSightEventPath=3oEatSzc ad.mrt=1438351373531 dmac=18:a9:05:6f:41:28
flexString2=Server ad.customerName=RCERT-C ad.customerExternalID=RCERT-C
modelConfidence=0 ad.eventAnnotationStageID=R9MHiNfoAABCASsxbPIxG0g\=\=
suser=NT AUTHORITY\SYSTEM ad.locality.i=1 eventAnnotationFlags=0
ad.dvcmac=27114219716904#015CEF:0|McAfee|Host Intrusion
Prevention|8.0.0|1148|CMD Tool Access by a Network Aware Application|Low|
eventAnnotationEndTime=1438351373531 eventAnnotationEventId=1
eventAnnotationAuditTrail=1,1438108536223,root,Queued,,,,
request=file:////opt/Oracle/Middleware/Oracle_IDM1/jdk/bin/java
assetCriticality=0 eventAnnotationModificationTime=1438351373531
ad.customerID=S7-ExhzUBABCBAeloelS4HQ
eventAnnotationStageUpdateTime=1438351373531 sourceGeoCountryCode=US
sourceGeoLocationInfo=Fort H slong=0 slat=0 sourceGeoPostalCode=85612
sourceGeoRegionCode=AZ dntdom=AMED destinationGeoCountryCode=US
destinationGeoLocationInfo=Fort H dlong=0 dlat=0
destinationGeoPostalCode=85612 destinationGeoRegionCode=AZ
```

[3] https://protect724.hp.com/docs/DOC-1072

2 Activities

The data generation script can create the following types of activity: web, SSH session, botnet, malicious redirect, and SMTP. Given a timestamp, each activity checks a set of conditions to determine if a new event should be generated at that time. When an event is generated, the script intelligently decides which formats should be written based on the type of activity (SiLK, Sourcefire, WCF, HBSS, or any combination thereof). For example, activity only seen on a host will be written to HBSS but not the other formats, whereas legitimate web activity will be written to SiLK and WCF, since the request will be seen by the web proxy and the traffic by SiLK.

Prior to analyzing the traffic it should be noted that even if an activity would have been blocked by an "in-line" control capable filtering traffic, we do not simulate the actual block. Therefore, WCF will record a threat record (which would normally result in a malicious web request to be blocked), but here we record the alert and act as if no block occurred (giving HBSS the opportunity to see an infection from a malicious download).

2.1 Web Traffic

The data generation script creates a new web activity every one to three minutes. The first step in a web activity event is a *DNS request* from client with a randomly assigned internal IP address to a hard coded external host listening on port 53. After the *DNS request* and corresponding *DNS response*, the data generation script creates between one and five *follow-on behaviors* using the same client IP and a randomly selected external IP address. Each *follow-on behavior* requests different URLs with the same domain name. These *follow-on behaviors* are separated by less than one second and the destination port is either 80 or 443, depending on whether the URL uses http or https. Once the *follow-on behaviors* are complete, up to two *ad behaviors* are generated. The URLs for these *ad behaviors* come from common advertisement domains with random strings concatenated to form a realistic looking adware URL.

Output Types

Output	Generate Records	What is captured
SiLK	Yes	Each DNS and web request and response
Sourcefire	No	Since this behavior represents normal activity no alert is generated.
WCF	Yes	Each Web Request
HBSS	No	Since this behavior represents normal activity no alert is generated.

2.2 SMTP

The data generation script creates a new SMTP event every one to five minutes. About 50% of events represent incoming behavior, with more packets flowing from an external IP to an internal SMTP server IP listening on port 25. The other 50% of events represent outgoing behavior, with more packets flowing from an internal client IP to a randomly selected external SMTP server listening on port 25. To simulate the strange manner in which some servers teardown TCP session, 35% of the events will contain the "SAR" flag; the other 65% will contain the "SAF" flag.

Output Types

Output	Generate Records	What is captured
SiLK	Yes	All SMTP activity that crosses the border
Sourcefire	No	Since this behavior represents normal activity no alert is generated.
WCF	No	Since this behavior represents normal activity no alert is generated.
HBSS	No	Since this behavior represents normal activity no alert is generated.

2.3 SSH

The data generation script creates an SSH event every five to ten minutes, then records that event in the SiLK output format. Each event contains the "SAF" flag and simulates an SSH session between a client using a random ephemeral port number and an SSH server listening on TCP port 22. The client initiates the TCP handshake with the SSH server in symmetric byte size. Once a secure connection is established, each session can last up to 30 minutes. Finally, the client terminates the connection with TCP, again with symmetric byte size.

Output Types

Output	Generate Records	Describe
SiLK	Yes	All SSH activity that crosses the border
Sourcefire	No	Since this behavior represents normal activity no alert is generated.
WCF	No	Since this behavior represents normal activity no alert is generated.
HBSS	No	Since this behavior represents normal activity no alert is generated.

2.4 Botnet

When the data generation script is run, it creates a botnet that consists a randomly generated set of between 2 and 20 internal IPs, which contact an external command & control server listening on port 6667 (IRC) every 2 minutes. The bots act out-of-phase.

Output Types

Output	Generate Records	Describe
SiLK	Yes	All SMTP activity that crosses the border
Sourcefire	Yes	Since this behavior represents normal activity no alert is generated.
WCF	No	Since no web request is made, no activity is detected
HBSS	No	We assume that no malicious activity is detected by HIPS and the activity is free to communicate.

2.5 Malicious Redirect

Every four to six minutes, the data generation script creates a web activity event followed by a malicious redirect. The malicious redirect occurs when the client IP from the ordinary web activity is redirected to a malicious URL, which infects the client's computer. The malicious redirect behavior contains the "Malicious code download detected" alert. Once infected, the client then joins a botnet, where each bot contacts a command & control server every two minutes.

Output Types

Output	Generate Records	Describe
SiLK	Yes	The DNS, web, and botnet activity is all seen crossing the border.
Sourcefire	Yes	The malicious download is seen by the IDS.
WCF	Yes	All of the web requests are seen by WCF.
HBSS	Yes	The infection is seen by HBSS.