

SCRIPTING WITH RSA NETWITNESS CONSOLE AND AUTOMATION VIA API & SDK

Guy Bruneau GSE
Senior Security Consultant
IPSS Inc.
@GuyBruneau

ABOUT ME

- Senior Security Consultant @ ipss inc.
- Incident Handler @ Incident Storm Center
 - gbruneau@isc.sans.edu
- Experience: planned, deployed, and used RSA NetWitness 8.x to 11.3.x in medium and large enterprise



AGENDA

■ RSA NetWitness Console

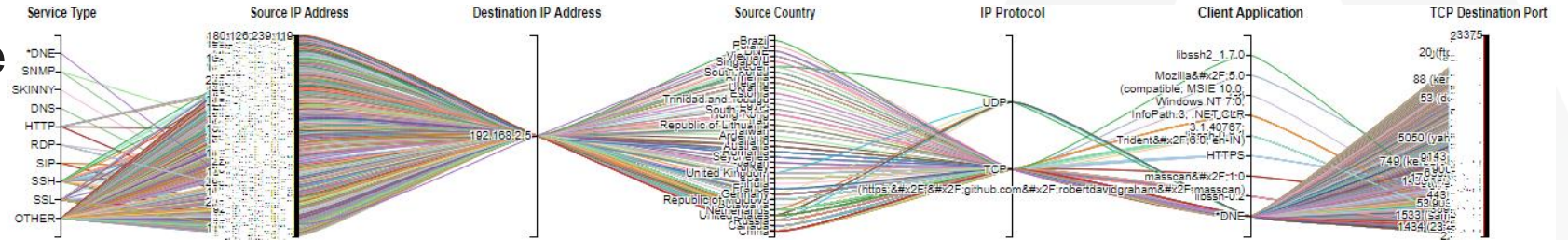
- Scripting for automation
- String & Regex search
- Exporting pcap from a metadata search
- Exporting pcap or files from a Session ID
- Post-Analysis: Importing large amounts of pcap data

■ API

- Loading feeds to multiple Decoders via a script
- Query metadata from other tools

■ SDK

- Query and save metadata to CSV



RSA NETWITNESS & COMMAND-LINE OPTIONS

- RSA NwConsole
 - Scripting to automate recurring jobs
 - Importing packets into a Decoder
- RSA NetWitness SDK search
 - String or Regex search
- RSA NetWitness SDK content
 - Extract SessionID to a pcap file
 - Extract all files from a SessionID to disk
- RSA NetWitness SDK with nwsdk_csv.py
 - Query meta of interest and save the result to a CSV file
- API
 - Upload multiple feeds to multiple Decoders via a script
 - Right-click query from a SIEM to RSA NetWitness Broker



AUTOMATION WITH RSA NETWITNESS CONSOLE

- Retrieve and save a list of files
 - EXE, ISO, ZIP, PDF, DOCX, etc
- Hash files of interest
 - Send result back to Log Decoder or SIEM of your choice
- Build script with the following options:
 - sdk open nw://admin:netwitness@concentrator:50005 (Broker:50003/Concentrator:50005)
 - sdk output /home/guy/executables
 - sdk content session=now-u where="filetype='zip','cab','exe'" render=files maxDirSize=1000000
- Script → NwConsole -f script &
 - Load <pathname> with a line delimited list of commands and execute them in order

```
> sdk open ...
> sdk output /home/guy/executables
New output directory set to /home/guy/executables
> sdk content session=now-u where="filetype='zip','cab','exe','pdf','docx'" render=files maxDirSize=1000000
17:39:47: Sessions 37692494 to 37692494 have meta range 2412346217 to 2412346217
17:39:47: Running in continuous mode...
17:39:47: New session range 1 to 37691310 has meta range 2412346218 to 2412266778
```



RSA NETWITNESS CONSOLE → STRINGS SEARCH

- RSA NwConsole
- sdk open nws://sdk:password@IP:port (Broker:56003/Concentrator:56005)
- sdk output /home/guy
- sdk search session=l-now where="\$META && \$TIME" search="keyword ='\$STRING' \$OPTIONS" pathname=\$OUTPUTDIR/\$FILE.txt"
- \$META && \$TIME
 - "netname = 'proxy dst' && streams = 2 && time='2019-09-16 19:29:36'-'2019-09-16 20:27:45'"
- \$STRING
 - SSH-2.0-PUTTY
- \$OPTIONS
 - sp cs nsm nds
 - search packets, case sensitive, do not search meta, do not decode session when searching

```
34264123 {SSH-2.0-PUTTY}b_S
34264253 {SSH-2.0-PUTTY}
34264439 {SSH-2.0-PUTTY}
34264464 {SSH-2.0-PUTTY}m0)
34264758 {SSH-2.0-PUTTY}{
34264881 {SSH-2.0-PUTTY}
34265680 {SSH-2.0-PUTTY}
34265760 {SSH-2.0-PUTTY}?*7S
```

RSA NETWITNESS CONSOLE → REGEX SEARCH

- RSA NwConsole
- sdk open nws://sdk:password@IP:port (Broker:56003/Concentrator:56005)
- sdk output \$HOME
- sdk search session=l-now where="\$META && \$TIME" search="keyword ='\$REGEX' \$OPTIONS" pathname=\$OUTPUTDIR/\$FILE.txt"
- \$META && \$TIME
 - "netname = 'proxy dst' && streams = 2 && time='2019-09-16 19:50:33'-'2019-09-16 20:49:44'"
- \$REGEX
 - SSH.*
- \$OPTIONS
 - sp ci nsm ds
 - search packets, case insensitive, do not search meta, decode session then search

```
37326728      SSH-2.0-Liquor<5&v-}kxscurve25519-sha256@lib{ssh.org,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1,diffie-hellman-group-exchange-sha1,diffie-hellman-group-exchange-sha256-v01@openssh.com,ecd}
37326731      SSH-2.0-Liquor<|.n9lycurve25519-sha256@lib{ssh.org,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1,diffie-hellman-group-exchange-sha1,diffie-hellman-group-exchange-sha256v01@openssh.com,ecd}
```

RSA NETWITNESS CONSOLE → IMPORTING PACKETS

- IMPORTANT! Imported pcap files retains original capture time
- Decoder configuration
 - Create a new Roles import_packets group on the Decoder
 - decoder.manage
 - sdk.manage
 - sdk.packets
 - Create *sdk* account with import_packets group
 - Decoder packet capture must be stopped
- Copy RSA NwConsole to a compatible workstation (i.e. CentOS 7)
 - cd to location of packets and login
- RSA NwConsole
 - login 127.0.0.1:50004 username password
 - import *.pcap

```
> login 192.168.25.33:50004 sdk netwitness
Successfully logged in to 192.168.25.33:50004 as session 4336
[192.168.25.33:50004] /> import *.pcap
```


RSA NETWITNESS CONSOLE → EXPORTING PACKETS

- RSA NwConsole
- Login to the Broker or Concentrator
- login 192.168.25.34:ssl username password (Concentrator)
- login 192.168.25.30:50003 username password (Broker)
- cd /sdk
- packets where= "\$META && \$TIME" pathname=\$OUTPUTDIR/FILE.pcap append=1
- \$META && \$TIME
 - "netname = 'proxy dst' && streams = 2 && time='2019-09-16 19:00:00'-'2019-09-16 20:00:00'"
- \$OUTPUTDIR = where to save the pcap

```
>login ...
Successfully logged in to 192.168.25.30:50003 as session 17968
>cd /sdk
[192.168.25.30:50003] /sdk
>packets where="netname = 'proxy dst' && streams = 2 && service = 22 && time='2019-07-12 21:10:48'-'2019-07-12 22:03:51'" pathname=/root/NWdata/root-charge2019-2019-07-12_22:03.pcap append=1
Writing packets to /root/NWdata/root-charge2019-2019-07-12_22:03.pcap (100%)
25 packets successfully written to /root/NWdata/root-charge2019-2019-07-12_22:03.pcap in 0 seconds (2.21 KB
0 B/sec)
Done processing, output in /root/NWdata/root-charge2019-2019-07-12_22:03.pcap...
Press ENTER to continue...
```

RSA NETWITNESS CONSOLE SDK → EXPORT PCAP/FILES

- Export a single or multiple SessionID as a pcap file
- Export all the files included in a Session ID, zip the result and password protect it
- RSA NwConsole
- sdk open nws://sdk:password@IP:port (Broker:56003/Concentrator:56005)
- sdk output \$HOME
- sdk content session=\$SESSIONID render=files
- Provide the SessionID to extract all the files
- Scripting with Linux to zip and password the files
 - zip -P infected files.zip files/*

```
> sdk output /home/guy/NWdata/files
New output directory set to /home/guy/NWdata/files
> sdk content session=38007748 render=files
17:26:34: Sessions 38007748 to 38007748 have meta range 2431670045 to 2431670158
17:26:35: Content has finished, the last session extracted was 38007749
17:26:35: Total number of sessions extracted: 1
17:26:35: Command finished in 0 seconds
Done processing, output in /home/guy/NWdata/files...
  adding: files/38007748-107-0_1.am_delta_patch_1.297.1005.0_3c822dea607d07167fb4fa3fb41d69370296e62c.exe (stored 0%)
  adding: files/38007748-107-0_2.am_delta_patch_1.297.1005.0_3c822dea607d07167fb4fa3fb41d69370296e62c.exe (deflated 30%)

Files have been zipped and password protected with infected and saved in /home/guy/NWdata
```

LOADING FEEDS IN AN AIRGAP NETWORK

- Upload multiple feeds to multiple Decoders
- Create an IP list for log/packet decoders
- FILES=/home/guy/feeds/netwitness
 - DLIST=/home/guy/scripts/decoder_list.txt
 - LLIST=/home/guy/scripts/logdecoder_list.txt
- curl -F file=@\$ {L} http://admin:netwitness@i:50104/decoder/parsers/upload/;
- curl -F file=@\$ {L} http://admin:netwitness@i:50102/decoder/parsers/upload/;

```
for i in `cat $DLIST` ; do

    cd $FILES
    for L in *; do
        curl -F file=@$ {L} http://admin:netwitness@i:50104/decoder/parsers/upload/;
    done
done
```

RSA NETWITNESS_SDK.SH SCRIPT

- Menu driven script
- Requires `nwsdk_csv.py` available at: <https://community.rsa.com/message/627004>
- Build to query and save the results from RSA NetWitness
 - Metadata or Payload Inspection → Using regular expression or string search
 - Using `nwsdk_csv.py` → search metadata output to CSV
 - Files → Export SessionID files included in payload
 - Packets → Export search query results to a pcap
- Configure script variables
- Can process file with list of IP, ports & everything else
- Copy available at:
 - https://handlers.sans.edu/gbruneau/scripts/netwitness_sdk.sh

```
All Meta, PCAP and Files are Saved in /root/NWdata

NetWitness Database Search Menu

1. Process a single query
2. Process a list contained in a file (ip or ports only)
3. Process a list contained in a file (everything else)
4. Web Traffic Analysis (Service=80)
5. DNS Traffic Analysis Summary (Service=53)
6. Extract packets to a pcap file
7. Process String or Regex Search
8. Dump Session ID data as a PCAP or Files

e. Exit script

What is your choice? █
```

RSA NETWITNESS SDK WITH NWSDK_CSV.PY

- Can be run by itself but ...
 - `python ./nwsdk_csv.py -c https://broker:50103/ -k "time,ip.src,ip.dst,service,tcp.dstport,alias.host,client,server,directory,filename" -w "alias.host begins update,report && filename='<none>' && directory='/' && query exists && query length 100-u" --no-count --gmtime`
- Called by `netwitness_sdk.sh` for items 1-5
- Output search in a CSV format
- Results with or without a count or activity time
- Three modes available
 - Search metadata as a single query
 - Search an IP or Port list contained in a file text file
 - Search a list with other type of metadata that requires quotes
- Web traffic (service = 80)
- DNS traffic (service = 53)

SCRIPTING WITH NWSDK_CSV.PY

- Query RSA NetWitness meta and output results to a CSV file hourly via cron
 - This example use DNS
- Configured with metadata of interest
- Data can be analyze with other tools
- Copy available at:
 - <https://handlers.sans.edu/gbruneau/scripts/dnsmeta.sh>

```
# Account used to query the metadata
NWSDK="/usr/local/bin/nwsdk_csv.py -c http://192.168.25.30:50103 -u sdk -p netwitness"

NOW=$(date -d "now" "+%Y-%m-%d %H:00:00")
HOUR=$(date -d "now" "+%h")
TODAY=$(date -d "TODAY" "+%Y-%m-%d")
YESTERDAY=$(date -d "yesterday" "+%Y-%m-%d %H:00:00")
LASTHOUR=$(date -d "-1 hour" "+%Y-%m-%d %H:00:00")
FILENAME=$(date -d "-1 hour" "+%H")

# Last hour query
TIME="time='$LASTHOUR'-'$NOW'"

# Check to ensure today's directory exist
OUTPUTDIR="/opt/Metadata/DNS/$TODAY"

if [[ ! -d $OUTPUTDIR ]]; then
    mkdir -p $OUTPUTDIR
fi

# Metadata keys part of the query
OUTPUT="time,ip.src,udp.srcport,ip.dst,udp.dstport,analysis.service,alias.host,direction,dns.querytype,"

# NetWitness query that check the our and date
# If the time is between 0-22, data is save in today's directory
# If the time is 23, data is will be saved in the previous day's directory

if [ $FILENAME != '23' ]; then
    $NWSDK -k $OUTPUT -w "$TIME && service=53" > /opt/Metadata/DNS/$TODAY/dns$FILENAME.csv
elif [ $FILENAME == '23' ]; then
    $NWSDK -k $OUTPUT -w "$TIME && service=53" > /opt/Metadata/DNS/$YESTERDAY/dns$FILENAME.csv
fi

# Change directory permission to be readable by everyone
chmod 777 -R $OUTPUTDIR
```

URL & API QUERIES

- API query from another device
- <https://192.168.25.5/1/navigate/query/email='gbruneau@ipss.ca'/date/2016-12-12T00:00:00Z/2016-12-13T00:00:00Z/>
- API to get malware
- <https://192.168.25.5/investigation/17/event/EventID> <https://nwsa/investigation/17/malware/event/80660528>
 - where EventID = Report number and 17 = SA device number

Q&A

■ My contact information

- gbruneau@ipss.ca
- gbruneau@isc.sans.edu
- @GuyBruneau
- <https://www.linkedin.com/in/guybruneau>

■ Posts & Projects

- https://isc.sans.edu/handler_list.html#guy-bruneau
- <https://handlers.sans.org/gbruneau>

■ RSA Global Summit 2014 → **Tips & Tricks To Achieve Ludicrous Speed**

- https://technodocbox.com/83629123-Network_Security/Deploying-security-analytics-tips-tricks-to-achieve-ludicrous-speed-guy-bruneau-gse.html

■ 2017 RSA Charge presentations → **Metadata Is Like Gold - Tips & Tricks To Mine It**

- <https://community.rsa.com/docs/DOC-83010>



THANK YOU

RSA[®] CHARGE

— 2 0 1 9 —