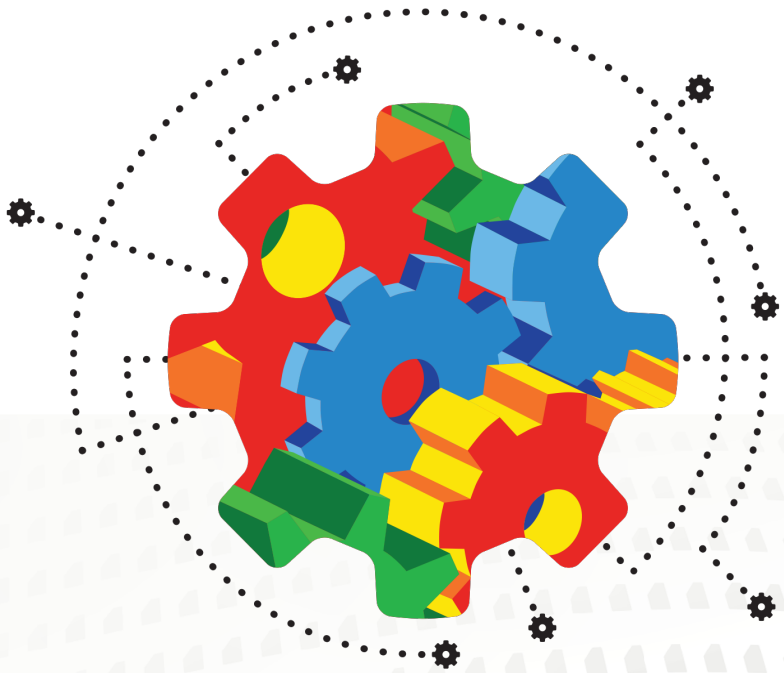
SOURCEfire®



Open Source Community Webinar

Costas Kleopa, OpenAppID Development Manager

Sourcefire is now part of Cisco.

Agenda

- Introduction to OpenAppID
- Use Cases
- OpenAppID Detectors
- Roadmap
- Community Announcements

OpenAppID – First OSS Application and Control

- OpenAppID Language Documentation
  - Accelerate the identification and protection for new cloud delivered applications
- Special Snort engine with OpenAppID preprocessor
  - Detect apps on network
  - Report usage stats
  - Block apps by rules
  - Snort rule language extensions to enable app specification
  - Append ‘App Name’ to IPS events
- Library of Open App ID Detectors
  - Over 1400 new detectors to use with Snort preprocessor
  - Extendable sample detectors

OpenAppID Language Documentation

- Lua JIT
  - History
- Application Detections based on Patterns in traffic
  - HTTP, SSL, SIP, RTMP/RTSP
- Other capabilities
  - Future Flow support
  - IPv6 supported

OpenAppID in Snort

- ./configure --enable-open-appid
- ./make
- ./make install
- More Information on Snort.org Blog
  - http://blog.snort.org/search/label/openappid

Use Cases

- Detecting Apps on Network and for IPS events
  - "(Event)",sensor_id="0",event_id="220",event_second="1394575648",event_microsecond="689239",sig_id="18760",gen_id="1",revision="4",classification="0",priority="0",ip_source="10.6.12.54",ip_destination="10.5.56.170",src_port="56148",dest_port="22",protocol="6",impact_flag="0",blocked="0",mpls_label="0",vland_id="0",policy_id="0",**appid="ssh"**
- Report usage stats
  - statTime="1394575530",appName="ssh",txBytes="2534",rxBytes="3234"
- Block apps by rule language extension
  - **drop** tcp any any -> any any (msg:"blocking any for ssh traffic"; appid: ssh; sid: 10000; rev:4;)
  - **reject** tcp any any -> any any (msg:"blocking any for sip traffic"; appid: sip; sid: 10001; rev:4;)

OpenAppID Detectors

- http://www.snort.org/snort-downloads/
  - Snort-openappid-detectors.tgz
- Latest Open Source Detectors
  - 1,400+ applications
    - Application Protocols
    - Web Applications (Payloads)
    - Client or Server apps
- Configuration
  - preprocessor appid : app_stats_filename appstats-unified.log, app_stats_period 300, app_detector_dir /usr/local/openappid/applications/
  - /usr/local/openappid/applications/custom
  - /usr/local/openappid/applications/odp

Detectors Deep Dive

- ./odp/appMapping.data
  - 629 Facebook 0 161 17 ~ facebook
- ./custom/*
- ./odp/lua/*
- ./odp/ports/*

Custom Detector Creation

./custom/stanford_university.lua

```
require "DetectorCommon"
local DC = DetectorCommon

DetectorPackageInfo = {
    name =  'stanford',
    proto =  DC.ipproto.tcp,
    client = {
        init =  'DetectorInit',
        validate = nil,
        clean =  nil,
        minimum_matches =  0
    },
    server = {
        init =  nil,
        validate = nil,
        clean =  nil
    },
}
```


Custom Detector Creation

```
function DetectorInit(detectorInstance)
    gDetector = detectorInstance;
    gAppId = gDetector:open_createApp('stnfrd');

    if gDetector.open_addUrlPattern then
        gDetector:open_addUrlPattern(0, 0, gAppId, "stanford.edu", "/", "http:");
        gDetector:open_addUrlPattern(0, 0, gAppId, "gostanford.com", "/", "http:");
    end

    gAppId = gDetector:open_createApp('stnfrd_resrch');

    if gDetector.open_addUrlPattern then
        gDetector:open_addUrlPattern(0, 0, gAppId, "stanford.edu", "/research", "http:");
    end

    gAppId = gDetector:open_createApp('stnfrd_stdnts');

    if gDetector.open_addUrlPattern then
        gDetector:open_addUrlPattern(0, 0, gAppId, "stanford.edu", "/gateways/students", "http:");
    end
end
```


Other Pattern Based AppID Detections

HTTP

```
-- NBC
gDetector:open_addUrlPattern(0, 0, gAppId, ”nbc.com", "/", "http:");
```

SSL

```
-- Facebook
gDetector:addSSLCertPattern(“0”, gAppId, ”facebook.com”);
-- VMWare
gDetector:addSSLCnamePattern(“1”, gAppId, ”VMware vCenter Server Certificate”);
```

RTMP/RTSP

```
Same as HTTP Host Patterns
gDetector:addRTMPUrl(0, 0, gAppId, ”espn.go.com”, “/”, “http:”);
```

SIP

```
—- Asterisk PBX detector
gDetector:addSipUserAgent(gAppId,””,”Asterisk PBX”);

—- text+ detector
gDetector:addSipServer(gAppId,””,"gogii.com”);
```

RAW Packets

```
gDetector:memcmp(gPatterns.first_tcp[1],#gPatterns.first_tcp[1],gPatterns.first_tcp[2])
```


Full API Documentation

http://www.snort.org/docs

Available Late Spring

Unified Tools

- u2openappid

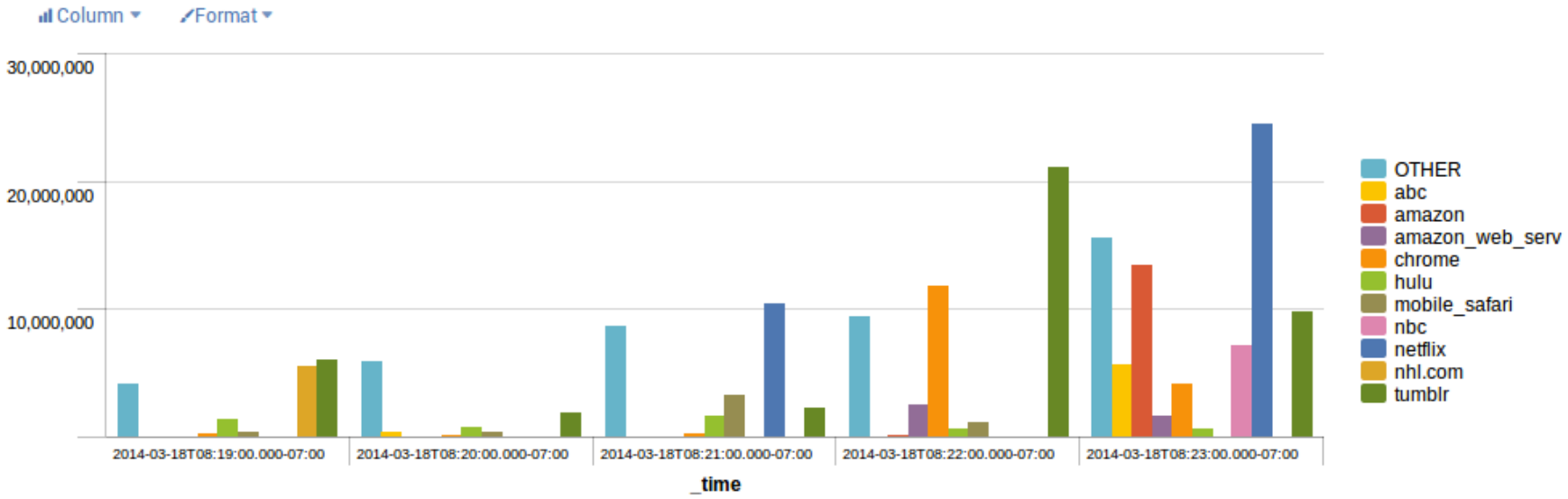
  statTime="1394575530",appName="mdns",txBytes="534",rxBytes=”230”

- u2streamer
  - Streams alerts or application statistics to syslog using auth:alert facility.
  - Logs can be directed to specific file, socket or remote hosts.
  - Uses a bookmark file to maintain continuity across restarts.
  - Example

```
u2streamer –path=snort.logdir –name=alert.file
```


Data Visualizations

- Format compatible with 3rd party tools



Roadmap

- New Pattern API's for
  - HTTP
    - Detection based on URL Parameters
    - Combinations of multiple HTTP Headers
    - New HTTP Headers support
  - DNS
    - Detection based on DNS Host
- More Detectors

Community Announcements

- snort-openappid@lists.sourceforge.net
- Detector Support
  - Share your new detectors
  - Issues or questions with the released detectors
  - Make new requests
- New Training Sessions
- Surveys

Questions