Cyber Security Challenge Australia 2017

Detect and Defend Solutions

Sponsored By

Australian Government

HackLabs

CISCO

IT'S HOW WE CONNECT

pwc

Microsoft

splunk>

facebook

CommonwealthBank

BAE SYSTEMS

Table of Contents

Challenge 1: OZMA

Challenge Description

Our engineers love sharing stories about Tik-Tok but who else read it?

Determine the MAC address of the 3rd party who intercepted traffic between two hosts. A full pcap of all hosts on a network segment is provided.

Designed Solution

Players use Wireshark or similar to analyse the provided PCAP. First step is to identify the the two hosts which are sharing information and then secondly identify the third host that intercepts this information. (MITM). The flag is the MAC of the third host.

Writeup

Using Wireshark to search on the text “OZMA” in the packet data, we should quickly identify the two hosts. (OZMA is a story from the Wizard of Oz). An example frame is below.

```
> Frame 14236: 2134 bytes on wire (17072 bits), 2134 bytes captured (17072 bits) on interface 0
v Ethernet II, Src: Vmware_91:2d:31 (00:50:56:91:2d:31), Dst: Vmware_91:ab:59 (00:50:56:91:ab:59)
  > Destination: Vmware_91:ab:59 (00:50:56:91:ab:59)
  > Source: Vmware_91:2d:31 (00:50:56:91:2d:31)
    Type: IPv6 (0x86dd)
v Internet Protocol Version 6, Src: fe80::165c:7e74:bb0e:4a5a, Dst: fe80::f732:71e7:2ee7:e645
    0110 .... = Version: 6
  > .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00 (DSCP: CS0, ECN: Not-ECT)
    .... .... .... 1100 0100 0100 1100 0011 = Flow label: 0xc44c3
    Payload length: 2080
    Next header: TCP (6)
    Hop limit: 64
    Source: fe80::165c:7e74:bb0e:4a5a
    Destination: fe80::f732:71e7:2ee7:e645
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
v Transmission Control Protocol, Src Port: 6000, Dst Port: 55048, Seq: 1, Ack: 1, Len: 2048
    Source Port: 6000
```

Some interesting points to note.

- TCP Port 6000 being identified as X11 is a red herring. Ncat was used on that port to transfer the text file.
- The OZMA text file was transferred twice - once without intercept and once with intercept.
- A key clue is the transfer is using IPv6 link local addresses - that should lead us to look for any link layer attacks.

Looking through the pcap data for IPv6 link layer protocol packets (RA, NS, NA etc), we will come across a number of a number of NS, NA sequences. Tracing those through, we will find the third host trying to poison the neighbour cache of the two legitimate machines. As the legitimate hosts also respond to NS messages in addition to the attacker, it is a race condition and we can see it takes multiple attempts before the MITM attack is successful.

In summary, this is the IPv6 equivalent and IPv4 ARP poisoning which will help us answer the Explain This component.

Flag

00:50:56:91:30:4f

Challenge 2: OZMA Explain This

Challenge Description

Explain this - How could you prevent this attack?

Designed Solution

Once we have identified that challenge 1 was an IPv6 link layer attack against the neighbour discovery process, there are two general approaches to preventing (or at least minimising the risk) this attack. Mention both approaches for full score.

Writeup

There are two general approaches – network based and host based. The "best" answers are highlighted in green. Answers that we may find from RFC6980 that won't work for the specifics of challenge 1 are highlighted in red but included for completeness.

Approach 1. Network based (as mentioned in RFC6980) includes

a. Use Static Access Control Lists (ACLs) in switches

b. L2 Filtering of ND packets such as RA-Guard (RFC6105) (RA-Guard would not help this CySCA scenario)

c. Neighbour Discovery monitoring tools such as NDPMon and ramond. ND monitoring capability is inherent in some switches. Cisco terminology

is IPv6 Address Gleaning to build an IPv6 to mac binding table and then IPv6 ND

inspection to validate ND packets against the binding table.

d. Intrusion Prevention Systems (IPS)

Noting that a shortfall of filtering and monitoring solutions is when ND attacks are combined with IPv6 fragmentation and hence the proposal is for hosts to implement RFC6960 and not use / respond to fragmented.

Approach 2. Host based.

Implement SeND https://tools.ietf.org/html/rfc3971

Noting the challenges of host OS support and deployment issues.

Challenge 3: It's Tic Toc. Not Tic Tap!

Challenge Description

Search the IPFIX records in Splunk and name the IP leaking data.

Analyse IPFIX records and determine the IP address that appears to be used to exfiltrate data.

Designed Solution

IPFIX records were collected from a test network segment where one host was transferring data to an outside host. There is other general Internet traffic to make it a challenge to find this (exfiltration) traffic. These records were loaded into the Splunk instance for ease of searching and visualisation.

Writeup

Without a Splunk instance to describe the

Sequence="280915"; Template="256"; sourceIPv4Address="172.16.205.231"; destinationIPv4Address="203.0.113.62"; ipClassOfService="0"; ipDiffServCodePoint="0"; protocolIdentifier="17"; minimumTTL="63"; maximumTTL="63"; sourceTransportPort="42912"; destinationTransportPort="53"; tcpControlBits="0"; ingressInterface="2"; bgpDestinationAsNumber="--";

ipNextHopIPv4Address="172.16.203.12"; egressInterface="3"; octetDeltaCount="10440"; packetDeltaCount="120"; flowStartSysUpTime="122297918"; flowEndSysUpTime="122357926";

TO BE COMPLETED

172.16.205.231

Challenge 4: Pick up even more STIX

Challenge Description

The It's Tic Toc. Not Tic Tap! Challenge is to analyse IPFIX records in Splunk to discover data exfitration. The data exfil was an outbound transfer of a file over DNS (dnscat2 was used)

Designed Solution

Create and submit a STIX 2.0 bundle with SDO's for the information you discovered in It's Tic Toc, Not Tic Tap!

Writeup

Once the malicious DNS traffic is identified, example IPFIX record of interest

Sequence="280915"; Template="256"; sourceIPv4Address="172.16.205.231"; destinationIPv4Address="203.0.113.62"; ipClassOfService="0"; ipDiffServCodePoint="0"; protocolIdentifier="17"; minimumTTL="63"; maximumTTL="63"; sourceTransportPort="42912"; destinationTransportPort="53"; tcpControlBits="0"; ingressInterface="2"; bgpDestinationAsNumber="--"; ipNextHopIPv4Address="172.16.203.12"; egressInterface="3"; octetDeltaCount="10440"; packetDeltaCount="120"; flowStartSysUpTime="122297918"; flowEndSysUpTime="122357926";

the following information can be described in STIX 2.0

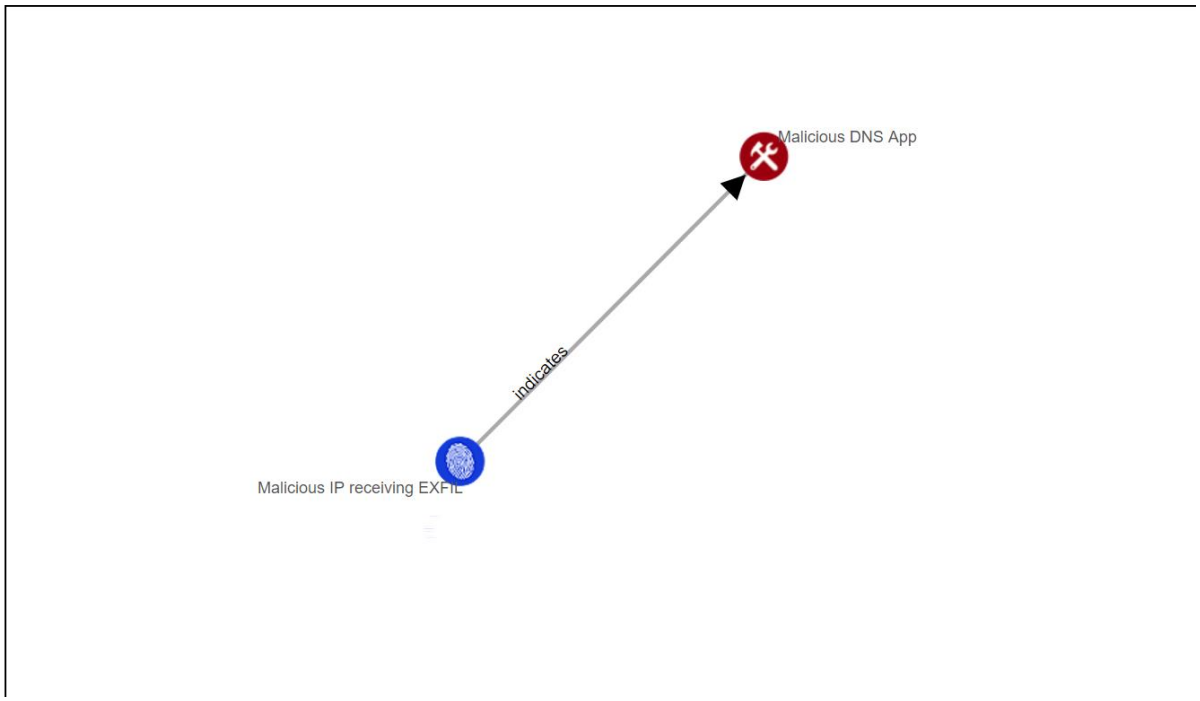
- The victim was 172.16.205.231 and the external receiving IP was 203.0.113.62
- The port / protocol was 53 DNS

An expected answer is to define an Observable based capturing the destination IP and Port and then define an Indicator for the malicious activity. Full marks to teams which define a STIX bundle including a relationship between two SDO. My example is showing a relationship between TOOL (DNS) and INDICATOR (DNS traffic to malicious IP).

Example STIX report below. Apart from visual inspection of the observable to validate the observable, the following visualisation tool will show any relationships in a STIX bundle.

https://oasis-open.github.io/cti-stix-visualization/

STIX Visualizer




```
{
 "type": "bundle",
 "id": "bundle--1234",
 "spec_version": "2.0",
 "objects": [
  {
    "type": "indicator",
    "id": "indicator--1234",
    "created": "2014-06-29T13:49:37.079000Z",
    "modified": "2014-06-29T13:49:37.079000Z",
    "labels": [
     "malicious-activity"
    ],
    "name": "Malicious IP receiving EXFIL",
    "description": "IP receiving exfil over DNS",
    "observables": [
```

```
    {
      "0": {
      "type": "ipv4-addr",
      "value": "203.0.113.62"
      },
      "1": {
      "type": "network-traffic",
      "dst_ref": "0",
      "protocols": [
      "dns"
      ]
      }
    }
  ]
},
{
  "type": "tool",
  "id": "tool--1234",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "labels": [ "network service"],
  "name": "Malicious DNS App"
},
{
  "type": "relationship",
  "id": "relationship--1234",
  "created": "2014-06-30T09:15:17.182Z",
  "modified": "2014-06-30T09:15:17.182Z",
  "relationship_type": "indicates",
  "source_ref": "indicator--1234",
```

```
    "target_ref": "tool--1234"
  }
 ]
}
```