THE UNIVERSITY OF WAIKATO
Te Whare Wānanga o Waikato

University of Waikato
Network Research Group

openLI

Static IP Intercepts

OpenLI Training: Chapter Thirteen

Shane Alcock
University of Waikato
New Zealand
shane.alcock@waikato.ac.nz

IP Intercepts

- When an LEA wants to intercept ALL digital communications
  - Web browsing
  - Email
  - Instant messaging
  - VoIP
  - Everything!

IP Intercepts

- Identity for IP is more complicated
  - The key information is the IP address assigned to the target
  - Aim is to intercept all traffic for the target's IP
  - But how do we know what IP maps to our target?

Complications

- IP addresses may be assigned dynamically
  - And they can change dynamically as well!
- Different protocols for assigning IPs
  - RADIUS, DHCP, GTP
- A target may have (and use) an IPv6 address
- IP addresses can be shared by multiple users
  - CGNAT, etc.

IP Identity

- OpenLI has to support multiple ways to express identity
  - This lesson: interception for targets with static IPs

Static IP Identity

- Network users are assigned a fixed IP address (or range)
  - Never changes
  - Probably “hard-coded” somewhere
    - Mapping of users to addresses in a database

Static IP Example Scenario

- We have received an IP intercept warrant
  - The intercept target is “Shane Alcock”
  - We sell them a fiber optic connection
  - In our internal user database, they have the username “salcock”
    - Assigned a static IPv4 address: 10.1.18.217

REST API for IP Intercepts

- Adding a IP intercept via REST
  - POST request
  - Content Type is a JSON object

http://<PROVIP>:<RESTAPIPORT>/ipintercept

REST API for IP Intercepts

- The JSON object for our static IP intercept
  - Many properties are shared with VoIP intercepts

```
{
    "liid": "STATIC002",
    "authcc": "NZ",
    "delivcc": "NZ",
    "mediator": 1,
    "agencyid": "mocklea",
    "starttime": 0,
    "endtime": 0,
    "user": "salcock",
    "accesstype": "fiber",
    "staticips": [
        { "iprange": "10.1.18.217", "sessionid": 101 }
    ]
}
```

REST API for IP Intercepts

- The JSON object for our static IP intercept

```
{
    “liid”: “STATIC002”,
    “authcc”: “NZ”,
    “delivcc”: “NZ”,
    “mediator”: 1,
    “agencyid”: “mocklea”,
    “starttime”: 0,
    “endtime”: 0,
    “user”: “salcock”,
    “accesstype”: “fiber”,
    “staticips”: [
        { “iprange”: “10.1.18.217”, “sessionid”: 101 }
    ]
}
```

REST API for IP Intercepts

- The JSON object for our static IP intercept

```
{
    “liid”: “STATIC002”,
    “authcc”: “NZ”,
    “delivcc”: “NZ”,
    “mediator”: 1,
    “agencyid”: “mocklea”,
    “starttime”: 0,
    “endtime”: 0,
    “user”: “salcock”,
    “accesstype”: “fiber”,
    “staticips”: [
        { “iprange”: “10.1.18.217”, “sessionid”: 101 }
    ]
}
```

REST API for IP Intercepts

- The JSON object for our static IP intercept

```
{
    “liid”: “STATIC002”,
    “authcc”: “NZ”,
    “delivcc”: “NZ”,
    “mediator”: 1,
    “agencyid”: “mocklea”,
    “starttime”: 0,
    “endtime”: 0,
    “user”: “salcock”,
    “accesstype”: “fiber”,
    “staticips”: [
        { “iprange”: “10.1.18.217”, “sessionid”: 101 }
    ]
}
```

REST API for IP Intercepts

- The JSON object for our static IP intercept

```
{
    “liid”: “STATIC002”,
    “authcc”: “NZ”,
    “delivcc”: “NZ”,
    “mediator”: 1,
    “agencyid”: “mocklea”,
    “starttime”: 0,
    “endtime”: 0,
    “user”: “salcock”,
    “accesstype”: “fiber”,
    “staticips”: [
        { “iprange”: “10.1.18.217”, “sessionid”: 101 }
    ]
}
```

REST API for IP Intercepts

- The JSON object for our static IP intercept

```
{
    “liid”: “STATIC002”,
    “authcc”: “NZ”,
    “delivcc”: “NZ”,
    “mediator”: 1,
    “agencyid”: “mocklea”,
    “starttime”: 0,
    “endtime”: 0,
    “user”: “salcock”,
    “accesstype”: “fiber”,
    “staticips”: [
        { “iprange”: “10.1.18.217”, “sessionid”: 101 }
    ]
}
```

IP Ranges

- You can specify a subnet instead of a single address
- IPv6 addresses are fully supported
- You can specify multiple IP ranges if needed

```
{
    ...

    “staticips”: [
        { “iprange”: “10.1.18.208/28”, “sessionid”: 101 },
        { “iprange”: “2001:db8:abcd:0012::/64”, “sessionid”: 888 }
    ]

    ...
}
```

REST API for IP Intercepts

- Using curl to add the intercept on the provisioner

```
curl -X POST -H "Content-Type: application/json"
  -d '{
        "liid": "STATIC002",
        "authcc": "NZ",
        "delivcc": "NZ",
        "mediator": 1,
        "agencyid": "mocklea",
        "starttime": 0,
        "endtime": 0,
        "user": "salcock",
        "accesstype": "fiber",
        "staticips": [
            { "iprange": "10.1.18.217", "sessionid": 101 }
        ]
    }' http://172.19.0.3:8080/ipintercept
```

REST API for IP Intercepts

- Success can be observed in the collector logs...

```
openlicollector[110]: OpenLI: received IP intercept for target salcock from
provisioner (LIID STATIC002, authCC NZ, start time 0, end time 0)

openlicollector[110]: OpenLI: intercepting static IP range 10.1.18.217/32
for LIID STATIC002, AuthCC NZ
```

REST API for IP Intercepts

- And in the mediator logs

```
openlimediator[9279]: OpenLI Mediator: received "Activated" HI1 Notification
from provisioner for LIID STATIC002 (target agency = mocklea)
```

REST API for IP Intercepts

- Modify IP intercepts with PUT requests
  - Must include the LIID field in your JSON object
  - If modifying static IPs, include **ALL** ranges you want to keep
  - Other unchanged fields are optional

```
curl -X PUT -H "Content-Type: application/json"
  -d '{
        "liid": "STATIC002",
        "staticips": [
            { "iprange": "10.1.18.208/28", "sessionid": 101 },
            { "iprange": "2001:db8:abcd:0012::/64", "sessionid": 888 }
        ]
    }'
  http://172.19.0.3:8080/ipintercept
```

REST API for IP Intercepts

- DELETE and GET requests work exactly the same as VoIP

```
curl -X DELETE http://172.19.0.3:8080/ipintercept/STATIC002

curl -X GET http://172.19.0.3:8080/ipintercept/STATIC002

curl -X GET http://172.19.0.3:8080/ipintercept
```

Next Time

- Running our first IP intercept
- Inspecting IRIs and CCs from an IP intercept