Research update; 3rd SARNET meeting

Ralph Koning
Scenario

SARNET demo

Control loop delay:

By using SDN and containerized NFV, the SARNET agent can resolve network and application level attacks.

From this screen, you can choose your attack and see the defensive response.

Traffic layers

Toggle the visibility of the traffic layers:

Physical links  Traffic flows

Choose your attack

Start a Distributed Denial of Service attack from all upstream ISP networks:

UDP DDoS

Start a specific attack originating from one of the upstream ISP networks:

Origin: UNSELECTED – CLICK ON A CLOUD

CPU utilization  Password attack

Normal operation

Object information

nfvservices.as100

KIND

nfv

f31d8ba32-3c13-421b-baba-493e3679dc0b#img-nfv

xenoffXOLarge

uva-ml-w1

requestsActive

vwanlmsite/rf#vwanlmsite/Domain/vm

[yamaha elect johnson]

[10.100.4.100 10.100.4.101 10.100.4.102]

[ids honeypot:4.100|4.101|4.102]

13
Sampling

- Ringbuffer with n values (Default: n=10)
- New samples arrive asynchronously at about every 0.8 seconds (per metric)
- Samples for sales from two services are added together, worst case this takes about 1.6 seconds.

- Detect: 30 percent of the samples in the window are below or above threshold.

- Recover: if 70 percent of the samples in the window are above or below threshold.
Observables

- **DDoS attack**
  - Detected if: Abnormal UDP, Sales < thresh(200)
  - Recover if: Sales > thresh

- **CPU attack**
  - Detected if: CPU > thresh(85)
  - Recovered if: CPU < thresh, Sales > thresh

- **PWD attack**
  - Detected if: failed > ok OR failed > thresh(20)
  - Recovered if: failed < thresh

- **How to determine the right thresholds and observables?**
  - ML might help though has its caveats
  - Determining what observables are needed and which ones are important cannot really be automated, unless we have all the data.
Effectiveness
Determining effectiveness

- Take the samples for a observable
- Subtract the threshold for that observable
- Invert the samples when needed (for sales)
- Set all negative values to 0
- Use trapezoidal rule to determine integral
- Maybe normalize by dividing by baseline?
Partial recovery

Revenue
Baseline

Attack Start
Detect
Recovered
Max Recovery
Implement

Attack Stop
Partial recovery

- Current method: when variability does not exceed `<insert arbitrary value>` from the mean.
- Better? Regression line in sample window and look at the slope..
- How long should we wait to call partial recovery.
  - It can still fully recover… eventually?
Password attack
Captcha recovery

← Faster recovery
Timeout and scalability

• Timeout: After 120 seconds the chosen countermeasure fails.

• Scalability issue?
  – If it takes 2 minutes to try a single solution iterating over 30 solutions takes about an hour (worst case).
  – How about combined solutions.