

Blue Sensors, Sensor Correlation, and Alert Fusion

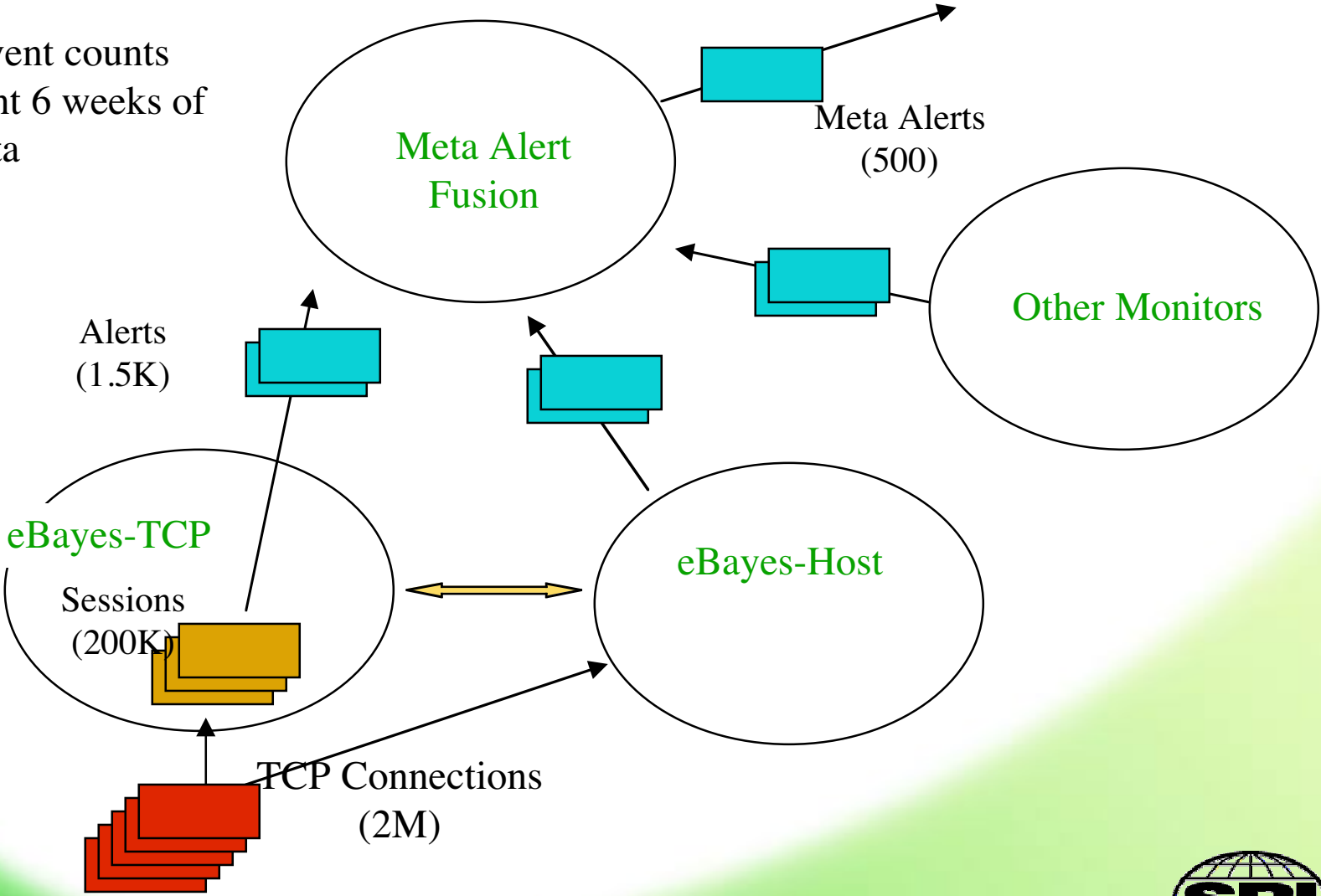
**Al Valdes
10/4/00**

EMERALD



Event Processing, Monitoring, and Correlation

Note: Event counts
Represent 6 weeks of
CSL Data



Functions of Correlation

- **Consolidate a large number of raw events into a single attack**
 - Emerald Alert Thread mechanism achieves this
 - Example: Syn flood may be thousands of events, reported as a single alert to the GUI
- **Use the state of one monitor to adjust another (for false alarm suppression or to enhance sensitivity)**
- **Fuse alerts (possibly from heterogeneous sensors) into one message**

Bayes Approach Succinctly

- **Start with a PRIOR expectation of the world**
- **Observe EVIDENCE**
- **Mathematically obtain an update belief in the state of the world, based on**
 - Prior
 - Observed Evidence
- **For long-running processes, the updated belief becomes your new prior expectation**
- **Would like to utilize external information to adjust prior**

“Blue Sensor”:

Service Availability Monitoring

- **Adaptively learns which services (ports) are valid on which hosts**
- **Based on Bayes analysis of traffic bursts, continuously maintains state for each such service**
- **Coupled with TCP session monitor:**
 - Effect on prior expectation of some feature values (e. g., successful 3-way handshake) mathematically reasonable
 - Attempts to connect to invalid services raise sensitivity: Stealth port scans in LL 99 data detected
 - Attempts by legitimate clients to connect to “down” services are now modeled with a different prior expectation: Effective false alarm suppression

Meta Alert Fusion

- **First, decide if the new alert matches an existing one**
- **Alerts contain a number of features**
 - Source(s)
 - Target(s)
 - Target port(s)
 - Type of attack
- **Approach here is to compare an alert to existing META ALERTS**
- **For each feature:**
 - Does it match? - Feature-specific *similarity*
 - Do we *expect* it to match? - Like a Bayes prior
- **We then compose an overall match. If it is good enough, we fuse the new alert with the existing one.**

So What is **Similarity**

- **Want a number between 0 and 1 (to manipulate like a probability)**
- **A Bayes belief fits the bill for some features**
- **For list features, how much do they overlap and how much could they overlap**
 - Example, eBayes-TCP does not see UDP ports.
 - If eBayes-TCP reports a portsweep, and
 - Another monitor reports a TCP and UDP portsweep, and
 - The TCP ports match, then the two lists overlap as much as they can

What is **Similarity (2)**

- **Another notion for lists: “Is contained in...”**
- **Specific to IP: “Is in the same subnet...”**
- **For attack type, sequencing information is more important:**
 - If we saw a probe, then our prior expectation of a subsequent DOS to a target of the probe goes up
 - In the case of a DOS, we expect source IP to be spoofed, so expectation of match for this feature is low.
- **So we would like to call a probe followed by a DOS to a target of the probe *similar* even if the source does not match**
 - Target of new alert “is contained in” target of meta alert
 - Expectation of similarity for source IP in a DOS is low
 - Expectation of DOS attack after a probe is moderate

Similarity and Anomaly Detection

- For each session, maintain list of ports accessed
- Compare to a database of lists
- If find a match, fuse the lists (new list is a superset of the two, with features for trimming rare entries)
- Otherwise, the new list is added to the database
- Return a database match (= anomaly) score. Flag very rare lists
- Result: nearly double the eBayes detections on the Lincoln data, with no false alerts
- On live traffic, anomalies are nearly half the alerts, but some appear to be good hits (example, stealthy scans to suspicious port patterns, NAPSTER, {80, 1080, 3128},...)
- Training interval is self-determined and much shorter than eStat in practice (minutes to hours vs. one month)

So Does It Work?

- 5 Days of live eBayes with the meta alert capability monitoring the CSL gateway (08/09 - 08/14/00)
- 131 alerts - 62 from anomaly detection
- Meta-alert capability reduces this to 68 at $SIM > 0.6$
- Subjectively, fused alerts make sense

Example of Fused Alert

- Saw numerous sweeps similar to:

```
Thread ID 4860 Class= portsweep BEL (class) = 0.994 BEL(attack)= 1.000
2000-08-10 06:50:28 from 193.230.37.2 ports 1268 to 32434 duration=
0.321
30 dest IPs: 130.107.1.1 130.107.3.1 130.107.4.1 130.107.5.1 130.107.6.1
130.107.7.1 130.107.8.1 130.107.9.1 130.107.10.1 130.107.11.1 130.107.12.1
130.107.13.1 130.107.14.1 130.107.15.1 130.107.16.1 130.107.17.1 130.107.18.1
130.107.19.1 130.107.20.1 130.107.21.1 130.107.22.1 130.107.23.1 130.107.24.1
130.107.25.1 130.107.26.1 130.107.27.1 130.107.28.1 130.107.29.1 130.107.30.1
130.107.31.1
6 dest ports: 635{30} 110{29} 143{29} 53{27} 21{27} 109{22}
count 164 max age count 0.16 code 3 svc 1 max-err 3.83 -opn 0 -oip
0 -oport 0
BEL 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
0.000 0.994 0.004 0.002 0.000
PCODE 0.000 0.000 0.000 1.000 0.000 0.000 0.000
SVC DIST 0.417 0.000 0.194 0.000 0.389 0.000

Non-sys alloc ports 6 port_anom 0.989744 code_anom 0.987017
Invalid hosts 30 Invalid ports 6 Neval 24 eval_count 0

LL-LIST 2000-08-10 06:50:28 130.107.1.1 to 130.107.31.1 portsweep 1.000
```

These Were Fused into the Following Meta Alert

Meta Alert Thread 15
Source IPs 193.230.37.2

Target IPs 130.107.1.1 130.107.3.1 130.107.4.1 130.107.5.1 130.107.6.1 130.107.7.1 130.107.8.1 130.107.9.1
130.107.10.1 130.107.11.1 130.107.12.1 130.107.13.1 130.107.14.1 130.107.15.1 130.107.16.1 130.107.17.1
130.107.18.1 130.107.19.1 130.107.20.1 130.107.21.1 130.107.22.1 130.107.23.1 130.107.24.1 130.107.25.1
130.107.26.1 130.107.27.1 130.107.28.1 130.107.29.1 130.107.30.1 130.107.31.1 130.107.1.2 130.107.3.2
130.107.4.2 130.107.5.2 130.107.6.2 130.107.7.2 130.107.8.2 130.107.9.2 130.107.10.2 130.107.11.2
130.107.12.2 130.107.13.2 130.107.14.2 130.107.15.2 130.107.16.2 130.107.17.2 130.107.18.2 130.107.19.2
130.107.20.2 130.107.21.2 130.107.22.2 130.107.23.2 130.107.24.2 130.107.25.2 130.107.26.2 130.107.27.2
130.107.28.2 130.107.29.2 130.107.30.2 130.107.31.2 130.107.1.3 130.107.3.3 130.107.4.3 130.107.5.3
130.107.6.3 130.107.7.3 130.107.8.3 130.107.9.3 130.107.10.3 130.107.11.3 130.107.12.3 130.107.13.3
130.107.14.3 130.107.15.3 130.107.16.3 130.107.17.3 130.107.18.3 130.107.19.3 130.107.20.3 130.107.21.3
130.107.22.3 130.107.23.3 130.107.24.3 130.107.25.3 130.107.26.3 130.107.27.3 130.107.28.3 130.107.29.3
130.107.30.3 130.107.31.3 130.107.1.4 130.107.3.4 130.107.4.4 130.107.5.4 130.107.6.4 130.107.7.4
130.107.8.4 130.107.9.4 130.107.10.4 130.107.11.4

From 2000-08-10 06:50:28 to 2000-08-14 01:47:18

Ports

Sum Hits 522.999 dot product 0.166816

Index 635 Prob 0.172815

Index 110 Prob 0.17024

Index 143 Prob 0.168917

Index 53 Prob 0.166216

Index 21 Prob 0.164595

Index 109 Prob 0.157217

Number of threads 26 Threads :4860 5564 6314 6980 7650 8223 8767 9287 9778 10350 10964 11577 12188 12634
13033 13802 14166 14525 17584 17907 18238 18558 18861 19185 19607 19980

Number of steps 1 Attack steps :portsweep

EMERALD



Meta Alert Content

- **IP Target list is superset of affected IP's**
- **Port list is similarly fused (note we get probability distribution over the ports)**
- **The date range is multiple days**
- **We retain the thread ID's of the component alerts**
 - Detailed drill-down capability
 - HTML Links in browse-able alert text file

Potential Usefulness

- **Correlating attacks that take place over extended time period (demonstrated)**
- **Potential to increase sensitivity**
 - Suspicious events below reporting threshold are nonetheless eligible for meta alert fusion
 - Over a sufficiently long time, collective suspiciousness rating rises to the reportable level
- **Scenario reconstruction**

Summary

- **Presented a 3-tier View of Alert Correlation:**
 - Threads for event aggregation
 - Modification of sensor state based on state of another sensor, or other external information
 - Alert fusion based on similarity measures
- **Introduced EMERALD System Availability Monitor (Blue Sensor)**
 - It state modifies session monitor, increasing sensitivity and suppressing false alarms
- **Defined Alert Fusion approach**
 - Based on appropriate similarity measures (some of which may be Bayes)
 - Couple this with expectation of similarity for each feature
 - Combine over features in common to determine if an alert matches an existing meta-alert thread
- **This has now been demonstrated in a meaningful sense on real data**