KoreLogic SECURITY

Shrinking the IDS Haystack

May 14, 2008

Presented by:
Hank Leininger

Co-Founder, KoreLogic
http://www.korelogic.com/

hlein@korelogic.com
BE5D FCCA 673B D18B 98A9 3175 896E 3D4A 1B4D C5AC

KoreLogic SECURITY

Agenda

- About Us.
- Don’t Believe the Hype.
- Reducing the Noise.
- Boosting the Signal.
- Free stuff.

KoreLogic SECURITY

Agenda

- **About Us.**
- Don’t Believe the Hype.
- Reducing the Noise.
- Boosting the Signal.
- Free stuff.

KoreLogic SECURITY

About me

- Background as a security practitioner.
- Now wear different hats as a consultant.
- Studying to be a grumpy old man.
- (Also, really bad at PowerPoint.)

KoreLogic SECURITY

About you

- How many practitioners in the house?
- How many people here sell security products?
- Who here writes code for a living?

KoreLogic SECURITY

Agenda

- About Us.
- **Don’t Believe the Hype.**
- Reducing the Noise.
- Boosting the Signal.
- Free stuff.

KoreLogic SECURITY

IDS is Dead.

A brief history of the IT security market[*]:

- 15 years ago the marketoids told us firewalls would save the world.

- 10 years ago the firewall was declared dead.

[*] I made this timeline up. Play along.

KoreLogic SECURITY

IDS is Dead.

Then we learned from our mistakes, right?

- 10 years ago the marketoids told us IDS would save the world.
- 5 years ago IDS was declared dead.
- For a while the market was excited by IPS, but that was declared dead too.[*]

[*] I'm skipping the rant about how proxy firewalls were technically superior to/safer than packet filtering, but lost in the market; now IPS and state-aware protocol-inspection capabilities with "helpers" ... make modern firewalls look more like proxies again.

KoreLogic SECURITY

IDS is Dead.

- Not to worry though, several new hype cycles have risen to take their place (just look around you…)
- And later it will become just as fashionable to disown all that is now new, and declare *it* dead.

KoreLogic SECURITY

Long Live IDS!

- For dead technology, there sure are a lot of firewalls out there. And you know what they're doing?

BEING USEFUL

- So too with IDS / IPS. Or at least, they *can* still be.

KoreLogic SECURITY

Long Live IDS!

- The proper take-away lesson isn't that IDS's aren't any good, it's that they're only good at what they're good at.
- Besides, you probably have some large number of dollars invested in IDS/IPS already. Wouldn't it be nice to get something out of them, rather than just check the audit box that says "yes, we have monitoring"?
- Granted, NIDS really *should* be dead, because all your network traffic should be encrypted, even internally. I'll check back in another 10 years, tell me how you're doing with that.

KoreLogic SECURITY

Agenda

- About Us.
- Don’t Believe the Hype.
- **Reducing the Noise.**
- Boosting the Signal.
- Free stuff.

KoreLogic SECURITY

Use a SIM, problem solved.

- There's a trend towards using a SIM / ESM to collect and correlate all events.
- That might work.
- But there are issues both at the low end and high end:
  - Good ones often too pricey for small shops, either in $ $$ terms, or labor to learn/support another tool.
  - Even the best fail at huge enterprises. More on that later.

KoreLogic SECURITY

The Wrong Way To Do It

- Let's revisit IDS for a minute. How is it best used?
- Well, here's how it's *worst* used:
  - Detecting attacks on Internet-exposed segments. Look Martha, an SMB worm!
  - Plug in, turn on, tune out. (Never looking at the IDS system again after setting it up.)

KoreLogic SECURITY

The Right Way…

- So that means we should:
  - Deploy on networks where we don't actually expect bad things to be happening every minute of every day:
    - Internal networks
    - B2B links / VPN links
    - "Private" WAN links
    - Intra-DMZ traffic
    - Between servers and SSL accelerators
  - Actually watch for stuff that happens
  - Watch… and *learn*!

KoreLogic SECURITY

As you wish.

- How could an IDS better serve you by learning?
  - First and foremost, false-positive reduction.
  - If you don't just ignore everything, the tendency is to disable noisy signatures.
  - Well, OK. But you've just expended energy and time to make this thing you paid for (or built, in the case of Snort boxes) do less. That seems sdrawkcab.
  - It's the right problem, but the wrong solution.

KoreLogic SECURITY

The new-cue-lurr option.

- Disabling signatures should be a last resort. (Other than ones that are really inapplicable to your environment, but you never enabled them in the first place, right?)
- Instead of disabling, always look for a way to tune by filtering out only the subset that is definitely irrelevant.
  - People grabbing /robots.txt off of a public website? Who cares?! But off of a private, non-published, requires-login-to-even-see-it website? That might actually be interesting.
  - Seeing fragments of shell scripts fly by on high ports? That could be trouble. Unless it's NFS traffic involving known fileservers and valid clients, in which case it's boring.

KoreLogic SECURITY

Less filling!

- So, we've become big fans of filtering.
  - Reduce false positives on high-severity alerts to near zero.
  - Reduce false positives for "background noise" events to cut down on the volume of data to be mined.
- The closer to the actual source of events or traffic, the better.
- Any human watching an IDS and looking at events is going to learn things about normal activity triggers false positives; why not teach the system not to bother you?

KoreLogic SECURITY

Tastes great!

- The overarching SIM/ESM mindset might say “Log em all, let God sort them out.” That is, record everything, feed it all to the SIM, and depend on the correlation engine to decide what’s interesting and what’s not.
- But there are problems with that.
- I love data. I *love* data. It is anathema to me to throw anything out. But sometimes, it's the right thing to do.

KoreLogic SECURITY

Redundancy all over again

- End devices still have to do all the work to generate events you're not interested in.
  - Lots of NIDS/NIPS are overworked, and it's a balancing act to keep them from running out of resources, especially as your network traffic grows over time. Why not cut them a break?
- The SIM / ESM still has to take in all this data, even after it has decided to suppress showing you the events.
  - For large enterprises, the volume of data that could potentially be sent to the SIM (millions of IDS events, tens of millions of firewall log entries per day) is beyond the capabilities of any amount of hardware you could justify in the budget.
  - The same problem occurs for smaller organizations, with both sides of the equation scaled down.
- GIGO. If you just throw all kinds of junk in, that's what you'll get out.

KoreLogic SECURITY

My Doctor Said Mylanta

- So while it's not mentioned in the glossy sales literature, it's almost always the case that you have to make painful trade-offs when choosing what data to feed in, or your SIM will choke to death.
  - Just firewall blocks (not accepts)?
  - Just login failures (not successes)?
  - Just high-priority IDS events (not the weird stuff that might have let you catch an 0day loose in procurement if you had noticed)?
  - The more informed your choices of what to ignore can be *before* data is fed to the mothership, the fewer categorical ignores you will need to have.

KoreLogic SECURITY

None shall pass.

- Oh, and what about IPS in particular?
  - No matter how aggressive people thought they might be with their IPS before they bought it, NIPS are almost always deployed to be pretty conservative in what they actually block.
  - All the more reason to provide feedback to the NIPS that says "*when you see this thing that looks like a buffer overrun in an RPC service, if it comes from our mainframe don't touch it*."

KoreLogic SECURITY

OK. Bloody smudge that might once have whinnied. Moving on.

KoreLogic SECURITY

Shrinking the Haystack

- **False-positive reduction in an IDS/IPS deployment:**
  - **Should be analyst-guided—nothing beats a brain.**
  - **Interface should be easy for power users to accomplish things quickly. Every redundant mouse click is a second of my life that I'm never getting back.**

KoreLogic SECURITY

Shrinking the Haystack

- IDS vendors should know which signatures trigger on various legitimate traffic. A huge percentage of events could be eliminated with some understanding of how default signatures behave in a real deployed environment:
  - DNS zone transfers between known/valid servers. (Anything not on the list is interesting.)
  - Port scans detected coming *from* busy webservers—they're replies to requests!
  - Sweeps of Internet hosts' port 80 coming from an internal proxy server. It's not a worm, it's ESPGooglBay!
  - Shell commands in high-port traffic to/from an NFS server. Hello, it's a UNIX fileserver, you're seeing shell scripts!
  - RPC traffic to/from NFS or NIS servers. As long as it's not sadmind…
  - Transfers of .EXE files from SMS / SUS servers to workstations.
  - Various SMB events occur all the time between domain members and Domain Controllers. But they shouldn't occur between one workstation and another...
- By applying these and similar "know your network" rules, we see between a 5:1 and 10:1 reduction in total IDS event load, and a 95% reduction in critical alerts requiring immediate attention.

KoreLogic SECURITY

Easier said than done

This is a lot to ask of the typical deployment, though:

- **IDS monitoring is seen as being labor intensive, and to only get more intensive over time.**
  - **And that's true, at least the way they are doing it.**
- **Really good IDS analysts are expensive.**
- **Therefore "Tier 1 monitoring" is often relegated to the cheapest NOC bodies money can buy, who escalate to Tier 2-3.**

KoreLogic SECURITY

Endless Escalation

And then what happens?

- A good deal of false positives are escalated; to try to keep the Tier 2-3 costs down, the Tier 1 folks get more and more recipes to follow to tune their escalation decisions.
- This is error-prone however, because they're Tier 1 people after all.
  - They don't have the background in either the enterprise or in network security to always get it right.
  - Plus, they have burnout jobs, and turnover is high, so there are always new bodies entering the meat-grinder pipeline.
  - The exceptions of course, rise to Tier 2 as quickly as possible, so again Tier 1 stays relatively starved.

KoreLogic SECURITY

Work Smarter, Not Harder

An IDS monitoring program with a good feedback loop would look like this:

- **IDS monitoring is done by high-end analyst(s), with junior people as *trainees*, not as monkeys.**
- **Every time the analyst looks into an event or group of events, if they are determined to be false positives, they provide feedback to the IDS system saying "These events were boring, I expect to see that kind of traffic between that server and these other servers."**
- **They never have to look at those false-positives again. The decision tree does not have to be amended and disseminated and digested by the junior staff, because the events just aren't there any more.**

(Here you might say a good SIM should be able to do that kind of thing for you, and I agree, yes, it *should*...)

KoreLogic SECURITY

As you wish.

- Sadly, this is a neglected capability in the IDS space.
  - Few IDS's have any capability to filter events in a really meaningful way, let alone an interface to manage such filters usefully.

- So, for some enterprise IDS deployments we help maintain, we wrote one.

KoreLogic SECURITY

As you wish.

- I'm going to talk a little about the tools we developed to manage filter lists on the Dragon IDS…
  - It should have come in the box!
  - It should be 10 times easier to use than what we grew!
  - I hope that nimble IDS / IPS vendors do add features like this. Yesterday would be nice!

KoreLogic SECURITY

DragonFilter: The Good…

Dragon's filter system:

- Dragon has supported event filtering since maybe 2000.
- Each event can have a BPF-style syntax filter attached to it. That's the syntax used by tcpdump, as in:

```
EVENTNAME tcp and host 4.5.6.7 and (dstport 111 or dstport 135)
```

- Supports arbitrary depths of nesting:

```
(a and b) or (c and (d or e or (f and g)))
```

- Some keywords supported are host, net, port; any can be 'src' or 'dst' variants (i.e. srcnet, dstport).
- You can describe some pretty complex structures this way. Great stuff!

KoreLogic SECURITY

DragonFilter: The Ugly

Dragon's filter system:

- You can edit these arbitrarily complex, possibly very long filter rules in a tiny one-line text box. Sigh. We were hand-maintaining rules that might be 500 characters long (back then, we thought that was big).
- And you can have an infinite number of typos and not know it until you try to push a policy, and then you have no idea what's wrong, just that you fail. (This has gotten better in recent versions.)
- We recognized the potential was there to drastically reduce the noise we had to sift through, but the poor user interface to filter management ate up much of the supposed labor savings.
- So we wrote wrappers that could do the job of building and maintaining the DragonFilter configuration.

KoreLogic SECURITY

vi is my IDE

Our filter management system:

- Written by geeks for geeks.
- It's some perl scripts, and a lex/yacc parser and syntax-checker.
- We've published the whole thing as opensource, GPLv2:
  - http://www.korelogic.com/tools.html
- The config file itself is valid perl code—this was the easiest way for us to have all the flexibility we could want.
- Uses revision control (CVS; could be SVN, git, etc) to track and annotate every change.
- Pushes rule changes into the Dragon management system for deployment to sensors.
- We get between 5:1 and 10:1, and in one case 50:1 reduction by applying these rules to production NIDS sensors.

KoreLogic SECURITY

Vendors, please build this!

- In case you hadn't figured it out: our system is *ugly*.
- I have no doubt the first requirement for a commercially viable form of this would be a GUI. A great many rules look kind of like this:

- I want some kind of topology building GUI. Let me define host objects, groups of hosts, and then each rule is the relevant groups and their relationships. Groups should be able to nest or inherit. Vendors, give me!

KoreLogic SECURITY

Agenda

- About Us.
- Don’t Believe the Hype.
- Reducing the Noise.
- **Boosting the Signal.**
- Free stuff.

KoreLogic SECURITY

Needs more cowbell

- We've been focusing mostly on the "noise" side of the signal-to-noise ratio.
- What about increasing signal?
  - Signatures custom to your environment, applications, and network protocols.
  - Signatures custom to your data.

KoreLogic SECURITY

Needs more cowbell

- Custom network-protocol signatures can be hard.
  - Reverse engineering network protocols is fun for only a few select crazy people.[*]
  - If you have some unusual but not unique networked applications, maybe someone else has already developed signatures (ask around).
  - Or, maybe you can find some boutique to develop some signatures for it (this is not a shameless plug).

[*] It is rumored that I am one such crazy person.

KoreLogic SECURITY

Needs more cowbell

- Application-specific signatures (using common protocols) are much easier.
  - Signatures that watch your application server talking to your database server. Your developers know what kind of queries should be made; how about signatures that look for “DROP TABLE” or “SELECT * FROM USERS;”?
  - Signatures that watch your LDAP directory server, knowing what kinds of lookups are expected, and trigger on others.

(This sort of thing is also doable with a good HIDS, but that's not what this talk is about.)

KoreLogic SECURITY

Honey Data

- Data-centric signatures are much easier.
  - Here's how it works. Think of it like dye packs in a bank teller’s drawer.
  - Seed your production databases with a few bogus records, and deploy IDS signatures that trigger any time those records go over the wire.

KoreLogic SECURITY

Honey Data

- For example, say you have an Oracle database with customer credit card numbers. Or maybe patient IDs.
  - Generate a few patient IDs that don't correspond to a real human, or some credit card numbers that aren't actually valid, etc.
  - Insert those into your databases. (You might have to shove them in manually, if the values you create are subtly invalid, and your usual front-end applications know not to allow them. That's actually a good thing.)
  - Now write IDS/IPS signatures that will trigger on those numbers, and deploy them on a sensor as close as you can get to your database.
    - Port 1521 signatures watching Oracle traffic
    - Port 80 or 7000 or whatever, your webserver<->appserver traffic

KoreLogic SECURITY

The call is coming from inside the house!

- If those signatures ever, *ever* trigger, you should be at instant red alert. That's a *whole lot* of signal.
- It's possible some developer did a "SELECT *" without a WHERE clause, or somebody is doing a backup over the network, that happened to trigger your alarm.
- But it's very, very likely that you are owned, you webapp has been compromised, and it's an attacker doing a "SELECT *" and downloading all your records. And they're doing it despite the firewalls and SIM and policies and procedures and oh God hurry up and pull the plug!

KoreLogic SECURITY

More Dye Packs

- This technique isn't confined to database records, although that's one of the best uses (and the most significant if they should ever trigger, because they indicate compromise, not just attack).
- Catch username harvesting: most IDS's have generic signatures for the ways in which attackers might harvest usernames. finger @, EXPN foo, GET /~bar/, etc. But apply the same concept here and you get more signal.
  - Watch for login attempts, finger, EXPN, etc for specific bait usernames.
  - If they trigger, not only do you have an attacker, but the attacker has already done their homework, had some measure of success enumerating users, etc.

KoreLogic SECURITY

More Dye Packs

- Data-centric signatures are *so* easy to do. IDS's *should* come with a set of default templates for Oracle queries, LDAP queries, let's say AD lookups, etc out of the box, with a tutorial or wizard that explains what you need to do, and generates deployable signatures once you fill in the blanks.

- But they don't, at least not yet, because trade rag reviewers wouldn't get it, and wouldn't give them any credit. So go tell PCWe^H^H^H^HeWeek you want this feature, so they'll ask vendors why they don't support it.

KoreLogic SECURITY

Agenda

- About Us.
- Don’t Believe the Hype.
- Reducing the Noise.
- Boosting the Signal.
- **Free stuff.**

KoreLogic SECURITY

Free stuff.

- I actually already gave a brief overview of the filter tool and mentioned its URL:
  - http://www.korelogic.com/tools.html
- This presentation will soon be available at:
  - http://www.korelogic.com/resources.html
- If we have enough time to get to this slide, I'll discuss the filter tool in a little more detail. I'll certainly take questions via email, etc.

KoreLogic SECURITY

Thanks!

- Any questions?
- Hank Leininger <hlein@korelogic.com>

  BE5D FCCA 673B D18B 98A9 3175 896E 3D4A 1B4D C5AC
- http://www.korelogic.com/
- http://marc.info/