iVoyeur

A Handful of Gems

DAVE JOSEPHSEN

Dave Josephsen is the author of *Building a Monitoring Infrastructure with Nagios* (Prentice Hall PTR, 2007) and is senior systems engineer at DBG, Inc., where he maintains a gaggle of geographically dispersed server farms. He won LISA '04's Best Paper award for his co-authored work on spam mitigation, and he donates his spare time to the SourceMage GNU Linux Project.
dave-usenix@skeptech.org

Fall is my favorite time of year. Not because of the leaves, or the pumpkins, or the other usual reasons (although the weather is nice). I love fall because fall is when the auditors leave. Beautiful fall, when they scurry back to whatever putrid swamps hatched them and I am free.

Just now, however, loathsome spring is beginning to settle in, and I close my window shades in dread of this my absolute least favorite time of year. Outside my window the pecan trees bud, the daffodils bloom, and I sense them stirring. Auditors across the country slithering to life inside the slick protective shells of their pupae, blinking their blind eyes and clutching their binders and clipboards close to their heart-like organs as they come fully to whatever passes as awareness for them.

Another 40 days, maybe less, and they'll descend on us like a blight. We'll mark their coming when they blot out the sun with appointments, reminders, and pre-engagement questionnaires. We'll know they're upon us when they dance from the shadows like marionettes, demanding answers to questions they don't understand. I'll be made to dance with them, these surreal pawns of an unseen overlord in a Shakespearian nightmare.

For months our dance will bring my professional life to a grinding halt. The dance will consume my productivity entirely, spinning us 10 hours a day up mountains of Word docs, and leaping us through fields of Dia diagrams [1], and all the while answering and answering and answering their strangely worded and contextless questions. Are my inputs validated? Do my authentications have two factors? Exactly how many bits do my encryptions have? Cha cha cha.

When they ask me what version of Active Directory I use, I will have to tell them that I do not use Active Directory. "Tsk tsk," they'll say to each other, marking things down on their clipboards. They'll exchange knowing glances and inform me of the need to remediate our lack of account management. I will then be forced to explain that we use OpenLDAP for account management. Their eyes will narrow, and they'll look at each other uneasily, suspicious that I'm making things up off of the top of my head. I know this will happen because it's happened every year for the past three, but the auditors have no memory. The auditors are freshly hatched each year and are not burdened by the weight of the past like you or me. The auditors don't need memories; they have clipboards and checkboxes instead.

That uneasy murmur in fact is about the only fun I'll have for the next few months. I know it's wicked of me, but I do so relish their discomfort when, for example, I reply "Snort" to their oddly phrased question about whether my networks contain IDS. A quirky answer befitting an odd question. Their pencils will hover above the checkbox, their indecision palpable. Is he making that up? No one would bring an IDS product to market called "Snort," would they? Unable to decide, they'll move on to the next question without marking down an answer just yet, only to be told that I update my IDS definitions with "OinkMaster." You should be there: it's delicious.

These tools—so lovingly created and whimsically named by my brothers and sisters in the nerdosphere—are like a handful of gems in my pocket. I pull them out and their brilliance dazzles the auditor hoards into confusion and dismay. I'm not working on any big monitoring projects at the moment, so I thought this month I'd share with you a few little gems I'll have in my pocket for the auditors this year.

Ganglia GWeb 2.0

My first article about Ganglia began with a lengthy rant about the sorry state of RRDTool display engines. The "spherical cow" [3] of RRDTool display engines has, in my mind, the following characteristics:

- It is polling-engine agnostic.
- It can read my RRDs from multiple directories, anywhere in the file system.
- It will show me graphs from any RRDs it can find without me having to configure anything.
- I can easily (in as few clicks as possible) tell it to create new graphs of aggregated data from the RRDs it knows about.
- It allows me to save these new graphs for later reference.
- It will provide a predictable URL to any graph it can show me, whether auto-detected or saved.
- It allows me to change the RRD options, such as size or date-range, of any saved or dynamically generated graph by modifying its URL.

Ganglia's polling engine and Web interface are designed to show graphs in the context of machine clusters, so the two will probably never be independent, but their ongoing Web interface redesign (dubbed GWeb 2.0 [2]) has addressed nearly every other one of my qualms. To the existing Ganglia Web interface it adds a search feature capable of displaying graphs matching regex search criteria, an aggregation feature that can aggregate data from any graphs that match a regex, and much more.

The displayed graphs now support specifying the time-range via a drag-box (e.g., Cacti [4]), and there are also text entry fields for specifying custom time ranges. Every graph I click on sends me to a linkable version of that graph. I can change the size, date-range, and myriad other RRD options (but not quite all of them) by modifying the URL parameters in these linkable graphs. This effort is really coming together nicely.

MonAmi

Speaking of separating polling engines from display engines, that same rant outlined some of the things I'd like to see in a hypothetically perfect polling engine:

- It speaks more than just SNMP.
- It provides sane defaults that are easily discovered and changed.
- It uses plug-ins to define data sources like SNMP daemons, external monitoring systems, and log files.
- If it stores data in RRDs, I can easily set whatever custom RRDTool attributes I want per device per metric.

I recently came across a fascinating little project called MonAmi [5], which seems to fit the bill. It aims to be a universal sensor network and all-around monitoring data middle man. MonAmi does not provide graphs or attempt to send notifications by design. Instead it is a pure data collection engine, using plug-ins to define data sources and endpoint destinations. It supports 16 data collection plug-ins, including plug-ins for the Apache Web server, Tomcat, and MySQL, and more generic system metrics such as file system, kernel process, and socket statistics.

Internally, MonAmi maintains data metrics in a local file system, and can aggregate and transmit the metrics it collects via transmission plug-ins to 12 different endpoint systems, including Nagios [6], Ganglia, MonaLisa [7], and GridView [8], as well as to log files and MySQL databases.

This, in my opinion, is an excellent design that wants only more plug-ins. Unfortunately, SourceForge lists its last code update as “2009-12-29,” and traffic on the forums appears to have died off in 2010. The code is stable, however; we’re currently using it on a few production hosts to move our Tomcat JVM metrics to Ganglia. I would really like to see someone pick up this little gem, polish it off, and love it.

JavaMelody

JavaMelody [9] is a stand-alone Java class implemented as a JAR file. It is easily integrated into your Web app by simply slipping it into the WEB-INF/lib directory and adding a few lines to the Web.xml. JavaMelody is about as lightweight as a JVM monitoring app can be. It doesn’t require that JMX be enabled, does not use profiling, and keeps no database of its own.

For your trouble you get 14 metric oodles of information about the running application; 28 application-specific metrics ranging from memory and CPU utilization to hit metrics such as HTTP sessions and active threads; and really interesting JDBC info such as SQL and Spring mean connection times. All of these are stored on the local file system in RRD format and graphed for you on a report page located at http://[servername]/monitoring.

But wait, that’s not all! The real-time Web-based report has a litany of text-based stats and information about the system itself, including process, thread, mbean, and JDBC connection viewers; JVM info including a full dump of all the class files and their versions; and tables of statistics about HTTP, Spring, and SQL connections.

I really wish JavaMelody would take the entire heap of this data and multicast it in XML format to gmond. It would be a windfall to have all of this centrally located in Ganglia. I’d dig into it myself, but I’m going to be busy for the next few months.

Snoopy Logger

If you have a centralized syslog infrastructure and you’re looking for a lightweight and easily manageable way to get an audit trail of every command executed on your Linux servers, Snoopy Logger is just the ticket [10]. Implemented as a shared library, which you place in ld.so.preload, Snoopy intercepts every call to execve() and logs the UID, PTY, and current working directory of every command executed to authpriv. Here’s a sample log line:

```
Mar 25 21:19:15 vlasov snoopy[12931]: [uid:1000 sid:5927 tty:/dev/pts/4 cwd:/
home/dave filename:/bin/cat]: cat /var/log/auth
```

If you're really serious you probably want something like GRSecurity [11] or SELinux [12], but in a layered approach, along with a file system integrity monitor, host-based IDS, and a centralized syslog infrastructure. It's an awesome little tool that does just exactly what it should, and with a name like "Snoopy" it's *bound* to go over with the auditors.

I hope you found something in there worth adding to your own bag of gems.

Take it easy.

P.S. Re-reading the intro, I realize that I may have been just a tad hyperbolic. The security auditors in this audience are, I know, just as frustrated with the ongoing security theatre as the rest of us and aren't to blame for any of it. The intro was intended to give you a smile, and also as a literary device to introduce some color into the subject matter.

P.P.S. Please don't flame me.

References

[1] Dia, a drawing program: http://projects.gnome.org/dia/.

[2] GWeb 2.0: http://ganglia.info/?p=343.

[3] Spherical Cows: http://en.wikipedia.org/wiki/Spherical_cow.

[4] Cacti Monitoring System: http://www.cacti.net/.

[5] MonAmi: http://monami.sourceforge.net.

[6] Nagios: http://www.nagios.org.

[7] MonaLisa: http://monalisa.caltech.edu/monalisa.htm.

[8] GridView: http://gridview.cern.ch/GRIDVIEW/dt_index.php.

[9] JavaMelody: http://code.google.com/p/javamelody/.

[10] Snoopy Logger: https://sourceforge.net/projects/snoopylogger/.

[11] GRSecurity: http://grsecurity.net/.

[12] SELinux: http://www.nsa.gov/research/selinux/.