Deploying ISS RealSecure in a Large Scale Environment Part 2: Manageability and Reporting

by Richard Reybok and Michael Engle

last updated Wednesday, April 26, 2000

Background

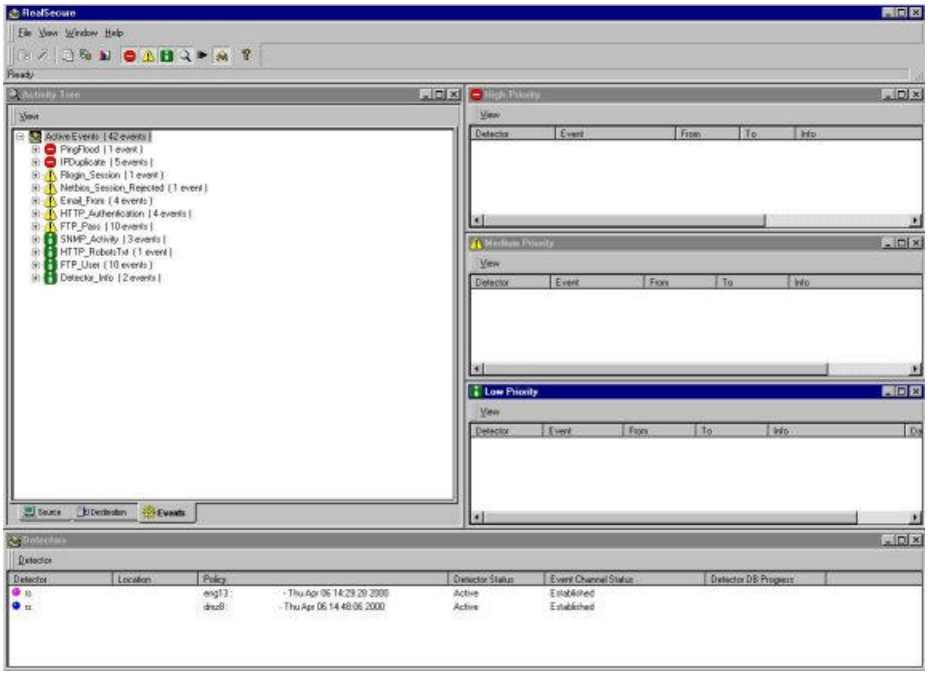
Welcome to the second half of our series on deploying ISS' RealSecure IDS product. The previous ar on some of the initial thoughts and practices of actually getting your infrastructure out there. This artic expand on that by helping you to manage your deployment, as well as to provide our ideas on event r executive level reporting. You will find this half to be much more detailed in terms of actual practical k can use today.

Manageability

When RealSecure is used in a smaller environments, it is easily managed. The console (Figure 2) all watch events in real time over an encrypted channel. Updating an agent or engine's properties is as s clicking and making changes. However, to deploy on a large scale you must take several steps into c

- Policy tuning - ensure you are not missing important events or gathering useless data
- Data collection - how to collect the data into a manageable format
- Event response - what to do if you detect hack attempts or a breach

Figure 2. The RealSecure Console (Click to view a larger image)



Policy Tuning

In a lab environment, it is easy to guess at what type of policy you require. However, once RealSecur on a production network your "default" policies will most likely need serious tuning. False positives wil up your database, causing queries to slow down and increase network traffic between engines, agen consoles. You will need to balance the data collection versus streamlining policies, as too many signa will leave you vulnerable to an unmonitored attack.

Enabling and Disabling Policy Signatures

There are several ways of tuning your policies. By enabling all signatures (or most, if you know there which do not apply) and watching the console you can get an idea of which events are important. Hov better test is to do the following:

Choose a heavily trafficked segment (engine) or heavily used server (agent) for testing.

1. Apply a liberal policy to the engine or agent.
2. Allow the machine to gather data for a period of time. You can monitor how many events are in machine's database by right clicking and selecting "maintain log" or using the `EngineMgr.exe` command.
3. Ensure the rsntclientlog.mdb file is empty on your Console machine. Note that "empty" does no deleted. ISS can provide you with a skeleton database file that just has the table structures in it have one. The file should be about 235k.
4. Perform a database synch with the console or EngineMgr.

You can now run the console's reporting tool to see what your top 20 events are. This will help you tri unnecessary signatures. After you have done this several times on different segments and on differer will be ready to push your policies out to your machines.

Different Policies for Different Environments

Unless you have a boring infrastructure, you are going to find you will be maintaining a number of diff files. Depending on a machine's role (e.g. File and Print, Domain Controller, Web Server) you will hav signatures enabled. You do not want to watch for scans of port 80 on a Web server because these pc always in use. Also, you will most likely want to monitor program execution on a NT domain controller necessarily on an application server. The same concepts hold true for network engines as well. Durin large-scale RealSecure deployment we needed nine policies.

Data Collection and Detector Management

As your systems are gathering data, it must be collected into a central repository. The local detectors (agents/engines) have a database that will fill over a period of time, and events will be lost. To prever data must be collected into a central database server on a regular basis. Unfortunately, a single cons associated database cannot handle having a very large (500+) number of detectors reporting to it. To ISS provides SAFEsuite Decisions as a central data warehouse facility. It uses a combination of a SQ data collection agents and reporting tools to collect and analyze the data. This gives you the ability to events over a greater period of time to improve your ability to provide both a greater number of report perform historical trend analysis.

Getting the data from the detectors to this reporting infrastructure is where the deployment is at it's mc challenging. A standard RealSecure deployment provides two methods of collecting data - The RealS Console (a GUI), and EngineMgr - a command-line interface. The console provides both real-time eve and data collection. EngineMgr only does policy, database and informational functions (no real-time n You will find you must use both of these if you are to have any hopes of maintaining a consistent and infrastructure that is flexible enough to allow quick policy modifications during crisis situations.

The RealSecure Console

The RealSecure console actively monitors each detector and pulls the remote database after a thresh hit. Unfortunately, it can be cumbersome to work with in a large environment. Every time the GUI is st detector must be added individually for monitoring/collecting. Also, the number of machines that each process is very limited in number. In an environment with 1000 detectors, you could need anywhere fr consoles just to collect the data. These numbers could vary depending on how heavily trafficked the c

and how powerful the console boxes are. This is due to both the high memory utilization of the consol the always-limiting 1GB limit of Microsoft Jet databases. It is also important to note that there is no pr correlating the databases of multiple consoles without the SAFEsuite Decisions product.

Another limitation in the current version of the console is the inability to push policies out to multiple m simultaneously. In order to update each machine, it must be selected, the policy selected, and then ap would be nearly impossible to do this to hundreds of machines in a timely fashion even with a staff of should be fixed in the next release of the product, but for now you must work around it.

EngineMgr.exe

This section explains in detail what we have found to be the most efficient and useful way to manage enterprise. By following what we explain here you should be able to manage 1000+ detectors from a s machine. The command line equivalence to the GUI is a programmed called EngineMgr.exe. Since it command-line program, it can be programmatically used to collect data or make changes. EngineMgr over 18 commands that mimic almost every function of the GUI. For the purpose of day-to-day operati following commands will be used:

- `-acqmaster`: Acquires master for a detector
- `-relmaster`: Releases master
- `-getdb`: Pulls data from the detector
- `-applypolicy`: Publishes a new policy
- `-clearlog`: Deletes the remote log (database) on the detector
- `-applyengineprop`: Defines the responses (email, SNMP, etc.) for a detector

Once you have policies in place, the -getdb option is to be used on a regular basis to collect data. Du RealSecure implementation we configured two consoles to pull data from 300 engines and agents. O was used for to poll network engines and critical agents. This totaled 51 machines - 22 engines and 2 They continuously gathered data and it was pulled into the SafeSuite database. The second console 250 agents in the same fashion. The total time required for a console to cycle through all detectors va 20 minutes to 2 hours depending on the time of day. During production hours more data was collecte time was necessary to pull. If the number of records held on the detectors is increased you could add console's responsible list.

Sample EngineMgr Algorithms

The following algorithms will give you an idea of what commands we ran to manage our environment. "$" symbols below represent an array and a variable, respectively.

Step 1 - Copy the console's public key, add the engine, and acquire master of all remote machines:

```
ForEach (@Servers) {
   Copy console's public keys to remote machine's KEYS directory
  EngineMgr.exe -a addeng -e $remote_ip
  EngineMgr.exe -a acqmaster -e $remote_ip
}
```

Step 2 - Apply your policies and response files. Note - you will need to use some type of logic to differ group of servers gets a particular policy. The Policy and Response files must reside in the C:\Program Files\ISS\RealSecure3.x\Policies directory on the machine running EngineMgr. We chose to use a de with different columns to organize the engines. Most of your machine will use the same response file email address, SNMP destinations, etc.) For example:

```
ForEach (@DomainControllers) {
  EngineMgr.exe -a applypolicy -e $remote_ip -p $dom_policyname
  EngineMgr.exe -a applypolicy -e $remote_ip -pr $responsename
```

```
}

ForEach (@DMZ_Host) {
  EngineMgr.exe -a applypolicy -e $remote_ip -p $dmz_policyname
  EngineMgr.exe -a applypolicy -e $remote_ip -pr $responsename
}
```

Step 3 - On a continual basis, pull data from the detectors into the local Jet Database.

```
:Start
ForEach (@Servers) {
  EngineMgr.exe -a applypolicy -e $remote_ip -db DSN=RSNTCONSOLE31 -dbdel T
  if size_of_mdb is > 800Megs {
      sleep 2 hours  #This gives SafeSuite Decisions enough time to pull data fr
      Replace the MDB file
}
goto start
```

Event Response

Aside from collecting data, you will need to react to certain events in a timely fashion. If you are watch console, you will have all the information you need. You also have the option of sending an email, SN message, or running an external program. If you use SNMP inside your company, you could use this f notification of important events. By the way, if you are an OpenView shop there is a nice plug-in that l to fit into the HP framework. Likewise an email could be sent to a pager or other address. For monitor networks, a TCP Kill can be used to terminate the session. One last option is to run an external progr could use this for just about anything - a proprietary notification system, configuring of network resour

If you choose to use the EngineMgr approach for managing your infrastructure, then you will most like deploying a great number of consoles. In this case, you will need to rely on a real-time event respons call this the full "console-less" deployment. We have found the best method is to configure your polici of your "high priority" events to this real-time system and then to generate daily reports on the rest of real-time system will hopefully be something that is sent directly to your pager or cellular phone, allow react immediately. This also applies even if you have consoles, but maybe not someone sitting in fron a 24x7 basis.

Reporting

RealSecure, when combined with SAFEsuite Decisions, allows you to analyze detector data in many f Some example report names include:

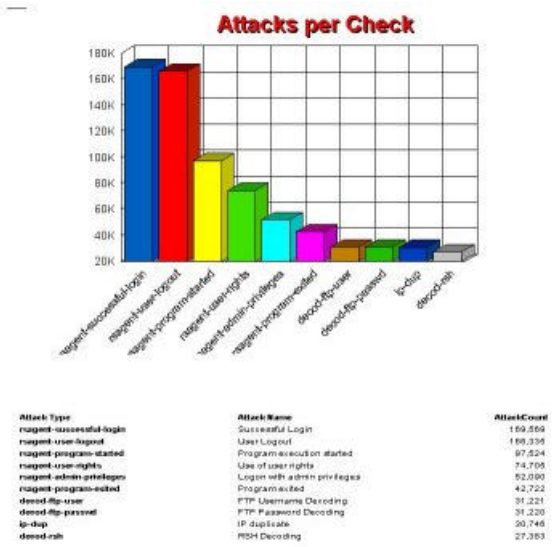
- Attack Analysis - Top 10 Common Attacks
- Attack Trends - Top 5 by hour. Includes separate options for weekend/weekday breakdown Co Attack from a single source
- Top 20 events
- Event Priority

There are over 25 reports included with Decisions. How you will use them depends on your particular and needs. By keeping previous reports you can compare different time periods to help identify new a

In addition to providing forensic data about attacks, overview reports such as "Attacks per Check" (Fig ideal candidates to give to other departments so they have an idea of the types of data being collecte the Security group visible, enhancing their image while not giving away any critical data that could be Often internal network and user account problems will show up on such a report and can be used by administrators to rectify the situation. When problems are solved, money is typically saved because n system performance is improved.

The best part about SAFEsuite Decisions is it is fairly standard in terms of interface options. You can your own reports using the Crystal interface. This allows you to really get specific reports based upon environment and your specific needs. As an example, we had a deployment where a daily report of al logons onto a specific set of Windows NT Terminal Servers was needed to help keep a tab on securi to supply data to the charge-back facilities. It is a flexible and almost necessary add-on to your deploy

Figure 3. A sample SafeSuite Decisions report (Click to view a larger image)



Other ISS Products

This isn't a sales brochure, but ISS also provide other security assessment tools which when used in with RealSecure greatly enhance your company's security:

- Internet Scanner - Identifies and addresses network vulnerabilities. Can perform network and sy from the view of a hacker.
- System Scanner - Identifies vulnerabilities at the software and operating system level and allow policies to be implemented.
- Database Scanner - Identifies vulnerabilities in your Oracle, MS SQL and Sybase databases.

When you combine all of these products into a single deployment, you can really get a complete secu monitoring solution from a single vendor. This allows you to be both pro-active and re-active in terms overall site security.

Known Issues, Tips & Tricks

There are several known challenges that can perturb someone deploying RealSecure in an enterpris main limitations of the software is its use of a Microsoft Jet database to hold data on the console. As v earlier, data is pulled from the detectors and stored into an MDB file. Due to limitations with Jet, this fi exceed 1 gigabyte in size. As data is collected into the MDB, the file must be replaced or purged on a basis. Some customers have implemented a method of pulling data directly into a SQL database via a DSN, but this is not supported by ISS at this time. Part of your `getdb` routine can check the file size, a sees it has reached a particular threshold it can be replaced.

Another issue is the lack of port options in EngineMgr. In order to communicate with agents and engir other side of a firewall, you must configure one port for each machine to communicate with it. For exa may configure port 1500 to communicate with web server 1, 1501 to communicate with web server 2, However, EngineMgr only allows you to use the standard ports so some type of port redirection must can usually finagle a "plug" type of gateway on your firewall to get around this. Again, your mileage m

One other consideration is the use of RealSecure Network Engines on a switched network. In order t traffic on a segment, port spanning must be done at the switch level. Your network operations group r involved and be part of the project plan before cooperating. If you have a deep switch fabric you may must deploy more engines or start creating additional VLAN's to replicate your data to. This could be engineering effort so plan for it in the beginning. This is not just an issue with ISS, but all network bas mentioned here for completeness.

Finally, again another general problem not one associated with RealSecure, is encrypted data. NO si be effective against network streams that are encrypted, such as HTTPS. The only chance you have host agent as only the destination host really sees the data in it's unencrypted format.

Conclusion

The market for intrusion detection systems is still in its early stages. If you have chosen to deploy ISS in your organization we think you have made an excellent decision. Be prepared to work with the soft expect greater functionality as it matures. With proper planning you can provide a first-rate level of pr your company.

Richard Reybok currently manages a Wall Street financial firm's global security engineering team where he puts his ninja security skills to t over six years experience in information security technologies and over ten years in the industry itself. When not defending the free world, R spending time with his lovely wife Jennifer and two beautiful daughters, Samantha and Kaitlyn.

Michael Engle currently manages computer security and incident response for a large financial firm on Wall Street. He has over 8 years exp industry. In addition to computing as a hobby, he enjoys scuba diving, skiing and traveling the world with his fiancée Diana.