Measuring Security

Jennifer Bayuk
ITG Security, Bear Stearns & Co. Inc.
115 South Jefferson Road, > Whippany, NJ 07981
(973) 793-5861 – Phone (973) 463-5388 – Fax Email: jbayuk@bear.com

I recently published a paper that compared several well-established methods of measuring security (See Computer Security Journal, Vol. XVII, No. 1, 2001). One observation in that paper was that all of these methods acknowledge the role of an "investigator." The investigator uses pre-defined criteria to assess the security of a given environment. The fact that an investigator may assign quantitative weights or values to his or her assessments does not change the fundamental qualitative approach. I recommended an automated approach.

The automated approach does not preclude weights and value judgement, it just requires that the qualitative judgements be made in advance. One pre-establishes formulas that will define security measurement, automates the measurement of the variables, then plugs the measurements into the formulas. This approach restricts the evaluative element of the security measurement process to formula-creation activity. It removes individual judgement from the measurement itself. The approach yields a quantitative measurement.

Of course, the hard part is to pre-establish the formulas. In the paper cited above, I suggested that IT Security follow a methodology that is successful in non-IT Security endeavors: a defect-elimination model. I defined security defects as corruption or intrusion. I defined corruption as the misconfiguration of mechanisms that prevent, detect, or facilitate recovery from harm to systems. I defined intrusion as the bypass of those mechanisms. I suggested many ways that evidence of corruption and intrusion could be automatically measured.

Regardless of what technology may be used to secure systems, this automated measurement of security defects may be applied. However, it is harder to apply for some security technology than others. The extent to which the security usefulness of a product can be measured should be a product evaluation criterion.

For example, suppose you are engaged in an evaluation of web access authentication products. There are core features of web authentication that you may require. These features might be:

- User ID identification
- Ability for users to choose from multiple authentication types
- Authentication via X.509 Certificate, Password, or hand-held token
- Ability to link User ID to existing authorization database
- Nondisclosure of User ID, authentication, and authorization data
- Full audit trail of User Administration activity, authentication, and authorization

Now suppose you have three competing products. Most evaluations would proceed with a spreadsheet that looks like this:

Criteria:	ProductA	ProductB	ProductC
User ID identification, business group distributed control over user setup, suspension, and termination	Yes	Yes	Yes
Ability for users to choose from multiple authentication types	Yes	Yes	Yes
Authentication via X.509 Certificate, Password, or hand-held token	Yes	Yes	Yes
Ability to link User ID to existing authorization database	Yes	Yes	Yes
Nondisclosure of User ID, authentication, and authorization data.	Yes	Yes	Yes
Full audit trail of User Administration activity, authentication, and authorization	Yes	Yes	Yes

There are "Yes" entries in all boxes for all vendors because often, the first pass at filling out this type of table is accomplished by talking to the vendor :-).

Then the formal product evaluation starts and the testers find out how the vendor accomplished each feature. They start making notes, shown in italics in the following table:

Criteria:	ProductA	ProductB	ProductC
User ID identification, business group distributed control over user setup, suspension, and termination	Yes, *need to restart software when business administrators change*	Yes, *but only by having multiple independent installations*	Yes, *need to give admin ID to person who sets up business administrators*
Ability for users to choose from multiple authentication types	Yes, *but cannot restrict choices*	Yes, *but can only be configured for one at a time*	Yes, *requires custom, signed object code*
Authentication via X.509 Certificate, Password, or hand-held token	Yes	Yes, *but X.509 Cert takes 10 seconds, hand-held token only authenticates to desktop*	Yes, *but only supports proprietary token device*
Ability to link User ID to existing authorization database	Yes, *but only supports Oracle*	Yes, *but only supports Progress*	Yes, *ODBC compliant*
Nondisclosure of User ID, authentication, and authorization data.	Yes, *all user and admin access via ssl*	Yes	Yes, *requires VPN software on client desktop*
Full audit trail of User Administration activity, authentication, and authorization	Yes	Yes	Yes

This is the typical path by which a product is chosen according to security requirements. The notes indicate that it will be easier to use some vendor products than others to accomplish the core feature set. Perhaps at this stage, one of the products may be eliminated.

But suppose in addition to verifying that the product satisfied security requirements, product evaluation teams would have to specify how they could verify that the features that satisfied the requirement were working as planned in production? The security "metrics" evaluation team has got to come up with metrics to show whether evidence of corruption and intrusion could be automatically measured. An example of the notes such a team would add to the above evaluation follows in bold:

Criteria:	ProductA	ProductB	ProductC
User ID identification, business group distributed control over user setup, suspension, and termination	Yes, *need to restart software when business administrators change,* **admin access required to read config, need to automate copy to monitor server**	Yes, *but only by having multiple independent installations* **can configure read config, can monitor config via mgmt software on monitor server**	Yes, *need to give admin ID to person who sets up business administrators* **must wrap admin menu or restrict via IP filter, need to automate copy to monitor server**
Ability for users to choose from multiple authentication types	Yes, *but cannot restrict choices* **authentication log does not show which method used, need enhancement**	Yes, *but can only be configured for one at a time* **log of user choice in proprietary format, not visible to admin**	Yes, *requires custom, signed object code* **must specify log requirements for custom code**
Authentication via X.509 Certificate, Password, or hand-held token	Yes **need to monitor config of CA and token server**	Yes, *but X.509 Cert takes 10 seconds, hand-held token only authenticates to desktop* **need to monitor config of CA, token server, and desktop**	Yes, *but only supports proprietary token device* **need to monitor config of CA and token server, need independent eval of token server**
Ability to link User ID to existing authorization database	Yes, *but only supports Oracle* **must add products' unique ID as field in existing database and keep synchronized**	Yes, *but only supports Progress* **allows db import/export of user names and passwords, will need to monitor all reads of associated files**	Yes, *ODBC compliant* **need controls over and usage monitoring of auth token stored in DB**
Nondisclosure of User ID, authentication, and authorization data.	Yes, *all user and admin access via ssl,* **but passwords and session cookies are stored in cleartext on operating system of web server, need to design and monitor OS file level security**	Yes **Uses private key encryption,where key is stored on every desktop, need enhancement request to detect intrusion**	Yes, *requires VPN software on client desktop,* **ODBC passwords in cleartext on Internal net, and admin can telnet into console port using app cleartext password, need to develop and monitor tunnel between servers**
Full audit trail of User Administration activity, authentication, and authorization	Yes, **but logs required for troubleshooting are in a proprietary format, offline reading of historical data requires separate product install, need to figure out if rollover and archive can be automated**	Yes, **but logs containing authentication activity are only included at debug level, which generates 1GB/day of non-security-related activity, also sent via syslog, need scripts to identify when logging has stopped and to rollover and archive**	Yes, **but direct console access to the operating system bypasses audit trail, need to monitor or block this channel and monitor block configuration, also logs are sent via snmp so need to integrate with Net Mgmt system**

Note that none of the features are left without comment. The measurement team must verify all security requirements in a way that does not depend on operating the product itself. It instead is dependent on what monitoring

and administrative processes are possible to verify that the security requirements are met. The comments indicate that those process exist in the IT environment and may be exploited to provide assurance the security requirements are met, or they indicate that a new process must be put in place.

Note that when each feature is assessed independently for teh ability to measure correct configuration and intrusion detection, it is common to find security product loopholes where features meant to satisfy one security requirement actually introduces vulnerabilities being measured with respect to another. The example of this above is Product C making use of ODBC compliant user ID databases where access to the database itself is not controlled.

In applying these requirements to security software deployment efforts, I have found many examples of huge, heavily funded software companies whose flagship security products:

- have no feature by which a user list can be exported to a non-proprietary format
- have no documentation that shows how configuration data displayed in the GUI corresponds to the configuration read into the product's software engine
- have no way to just log successful access attempts, just failed access attempts, or both
- allow backdoor cleartext passwords to administer the product via a network
- provide only unreliable and unsecure protocols for centralized log collection

Vendor response to my issues has been universal. They are following industry standards. We are left to conclude that industry standard security requirements do not yet include robust features by which we can verify that a product is correctly configured and/or is not being misused.

Yet, if automated measurement of security defects is to be applied, we must have these features. Security metrics should not be left to qualitative judgements based on investigative models. The extent to which the security usefulness of a product can be automatically measured should be a fundamental, not a secondary, security product evaluation criterion.