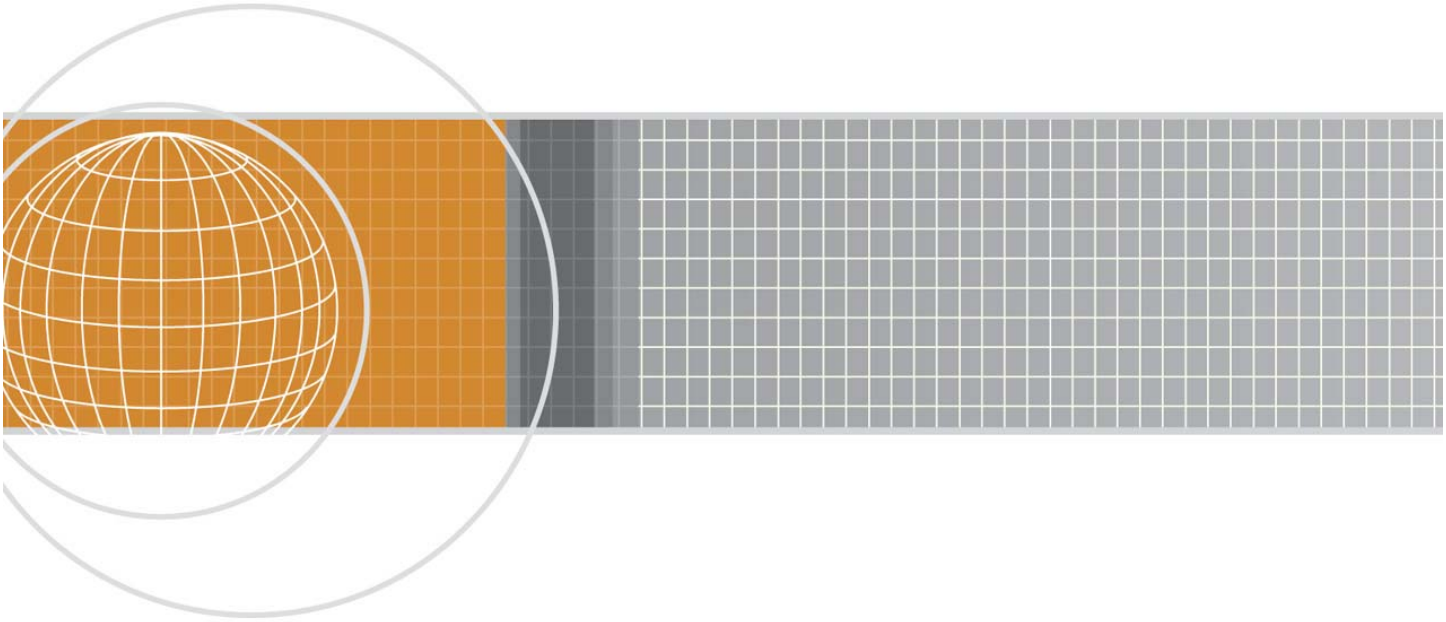
GUARDEDNET | Transforming Security Data *Into Knowledge*

A GuardedNet White Paper

The Practitioner's Guide to Incident Response Best Practices

By Ken Pfeil, CSO, Capital IQ



The Practitioner's Guide to Incident Response Best Practices

Far too often, companies put together an Incident Response Plan (IRP) and Incident Response Team (IRT) built primarily from reactionary instances. In other words, an incident had to occur for these very necessary and vital processes and plans to become a key ingredient of an organization's security defense posture. The basic premise of this guide is to assist the established IRT in becoming more efficient and effective in their existing processes and procedures and in planning for the future, and to re-introduce the information in a different light. This guide is not intended to be all-encompassing of every process, procedure, and way of responding to incidents. One short guide cannot do justice to the amount of information that needs to be considered for an effective IRP. This guide will explore the universal areas in which improvement can make the most immediate impact to an organization's overall incident response (IR) effectiveness.

The Key Ingredients of a Successful IRP

Every IRP has several "key ingredients" that constitute its overall makeup, and in turn, dictate to some degree its overall effectiveness and shelf life. These ingredients will no doubt vary in depth and function depending upon your organization's asset protection goals and security plan. One fundamental rule that should not be overlooked is that your IRP is an iterative and living process, which should be evaluated and tested periodically. The generally accepted rule of thumb for evaluation and modification is every 3 to 6 months. IR "fire drills" are just as important as Business Continuity Plan (BCP) testing, and in some cases, can coincide quite nicely with it. Some key ingredients of a successful IRP that should not be overlooked include:

- Baselines
- Threat identification
- Data collection and interpretation
- Response

The reason these can be considered key ingredients is that every process and procedure in your IRP will hinge directly upon how effectively these factors are established and implemented. Without a baseline, there is no way to establish "normal" events and traffic so that you can determine when you're experiencing "deviant" events and traffic. Without an effective threat identification mechanism(s) and process(es), the IR cycle breaks down. And without effective and efficient data collection and interpretation, it is difficult to determine when incidents occur that fall outside of your baselines, and thus makes it difficult to implement your IRP.

Baselines

The baseline you've established will ultimately encompass acceptable traffic patterns, thresholds, and times; frequency of certain events; correlation of events to systems or devices; desired (and realistic) response times for events; and so on. An incident baseline coincides with a particular point in time of your systems and directly correlates to business objectives, tolerances, and reactions to specific events. Because this information will change over time, you should think of your baseline as a moving baseline, as periodic adjustment will no doubt be needed. Some specific areas that can affect this baseline over time include:

- Introduction or removal of systems into the environment
- Acquisition or merger of a company or organization
- Changes in service providers

After you've established a baseline of normal traffic and events, comparison of and contrast to events outside of normal parameters becomes a much easier process. Anomalous behavior is also easier to identify with a baseline in place. The result of establishing a baseline is to provide a "yardstick" for which to measure future events by. A baseline is not necessarily a tangible object, but it is measurable and definable based upon applicable rules that your traffic must follow. These rules are defined based upon the normal needs of your environment and are tuned over time to reduce false alarm rates.

Threat Identification

A risk assessment can often be done to "kick start" threat identification, as quite often a risk assessment will reveal threats and situations to which you haven't given thought. A risk assessment is a process used to ensure that the security controls for a system (or systems) fully commensurate with associated risks. There are two types of risk assessment—quantitative and qualitative—and the several accepted risk analysis approaches fall into one of these two types.

Once you have performed a risk assessment, it becomes much easier to categorize potential attacks and scenarios that might not have an impact on your organization's operation. Specific risks, once identified, can also be correlated back to your original business impact analysis to see how these risks affect other areas of the business. Incidents that are not applicable to your environment should be logged, but filtered from normal view wherever possible (for example, hardware- or software-specific attacks directed at a platform you do not use). These attacks can be considered non-actionable, but are still logged mainly for pattern and trend analysis purposes.

Incidents are typically logged to a central log server. They are then fed to a database for trending, tracking, and subsequent tagging and categorization of the events. The database itself is typically contained on the log server to facilitate this process, or may be kept on another system as an additional security measure.

Threats can be identified from various sources such as:

- News and media
- Anonymous source or email
- Vendor notification
- Sudden increase in unclassified traffic
- Out-of-pattern behavior of systems

It is important to remember that even though you may have identified a specific threat to your environment, protecting against that threat might not be feasible in a particular situation. Not all threats are catastrophic in nature, and eliminating a mild threat may cause instability in some circumstances. For example, deleting a vulnerable Internet Information Server (IIS) mapping in order to mitigate an immediate threat before a patch is available would breaks the Web application on which you run your business; thus, the cure could be worse than the illness. In this situation, categorize the threat and tag it for monitoring instead of immediate action.

Data Collection and Interpretation

Data collection and interpretation involves the processes and mechanisms needed to collect and handle all relevant actionable items. Actionable items include log files, traffic captures, and so on. Interpretation of events can be a manual or automated process.

In the past, many people have shied away from automated processes out of concern that something critical might be overlooked. The advent of the intelligent correlation systems of today make these fears less justified and, in some cases, a moot point. An intelligent system that uses complex algorithms and is capable of "learning" traffic patterns makes the collection process more efficient. Either way you decide to go, data collection and interpretation will typically require

setting up a central log and management device to collect events. Be sure to restrict access to this system to only those who require it.

Incident Classification and Severity

To effectively respond to an incident, an effective classification mechanism is essential. To effectively classify an incident, threat identification should have been assessed. Table 1 shows an example incident classification matrix.

Category	Description	Action
Low	Events that cannot be definitively identified as attacks and have no effect on operations; for example: • False activation of intrusion detection systems • Isolated and non-repeated scans or pings from an external uncontrolled network • Malware detected and removed prior to being placed on a network	Log these incidents; mark them as incidents that do not require immediate attention.
Medium	Incidents that have no effect on operations and comprise identified but unsuccessful attempts to actively breach an information security policy; for example: • Accidental failure to physically secure systems overnight • Repeated active probes or port mappings from an external network • Malware that has been successfully contained or removed	Log these incidents; mark them as incidents that require attention (immediate attention not required, however).
High	Incidents made up of any successful attempt to actively breach an information security policy and might result in a minor or moderate effect on operations; for example: • Unauthorized access acquired • Malware found on more than one system or an inability to contain and remove the code • The defacement, alteration, or deletion of Web server files	Log these incidents; mark them as incidents that require immediate attention.

Table 1: An example incident classification matrix.

Incident Severity

Incident severity can be classified several ways. Some common severity classification schemas include low/medium/high and limited/moderate/critical, or you can classify incident severity by using an escalating numerical severity system such as:

- Severity Level 1—Small amount of abnormal or anomalous traffic or activity detected outside of the tolerance level of the established baseline; port scanning or probes detected on internal systems.
- Severity Level 2—Small amount of system probes or scans detected on external systems on which information is received relating to threats of which applicable systems may be vulnerable.
- Severity Level 3—A large number of systems are being probed or scanned; targeted exploits used, which may indicate reconnaissance has been done; penetration or Denial of Service (DoS) attacks attempted with no impact on environmental operations.
- Severity Level 4—Penetration or DoS attacks attempted with limited impact on operations; some risk of negative financial or public relations impact.
- Severity Level 5—Successful penetration or DoS attacks detected with a large impact on an organization's operations; significant risk exists for financial loss or public relations impact.
- Severity Level 6—Compromise of an internal host or device that is not outwardly facing; most likely an internal employee or insider.
- Severity Level 7—Compromise of an externally facing host or device.

In the case of a low/medium/high classification scheme, Severity Levels 1 and 2 would typically fit under Low, Severity Levels 3 and 4 would generally be classified as Medium, and Severity Levels 5 and higher would classify as High. Classification methods should not be approached as a "one size fits all" solution. Whatever classification method you choose to implement, make sure all incident scenarios relating to your business can be accounted for and classified. For instance, what if my credit card database is compromised? Are there any applicable laws in my location that directly affect my IRP (such as California State Bill 1386)?

Once you have developed a classification scheme within your organization, you can use the system as a tool to help determine where a threat falls within your IRP. This determination, in turn, will define which actions are appropriate for responding to the threat. For example, suppose malware is detected before it does any damage. Using the very basic matrix that Table 1 shows, the incident falls under the low category, so the action is to log the incident and move on. Thus, a classification scheme is an effective tool within your IRP.

IR Concepts and Procedures

A typical IR cycle consists of alerting, investigation, response, recovery, and lessons learned phases. The names and depth of these cycles may vary in your plan, but the concepts remain essentially the same. You can clearly see all of these aspects in one form or another in the sample IR flow diagram in Figure 1. (Bear in mind that this figure is simply an example, not a definitive illustration of all scenarios that your business might encounter.)

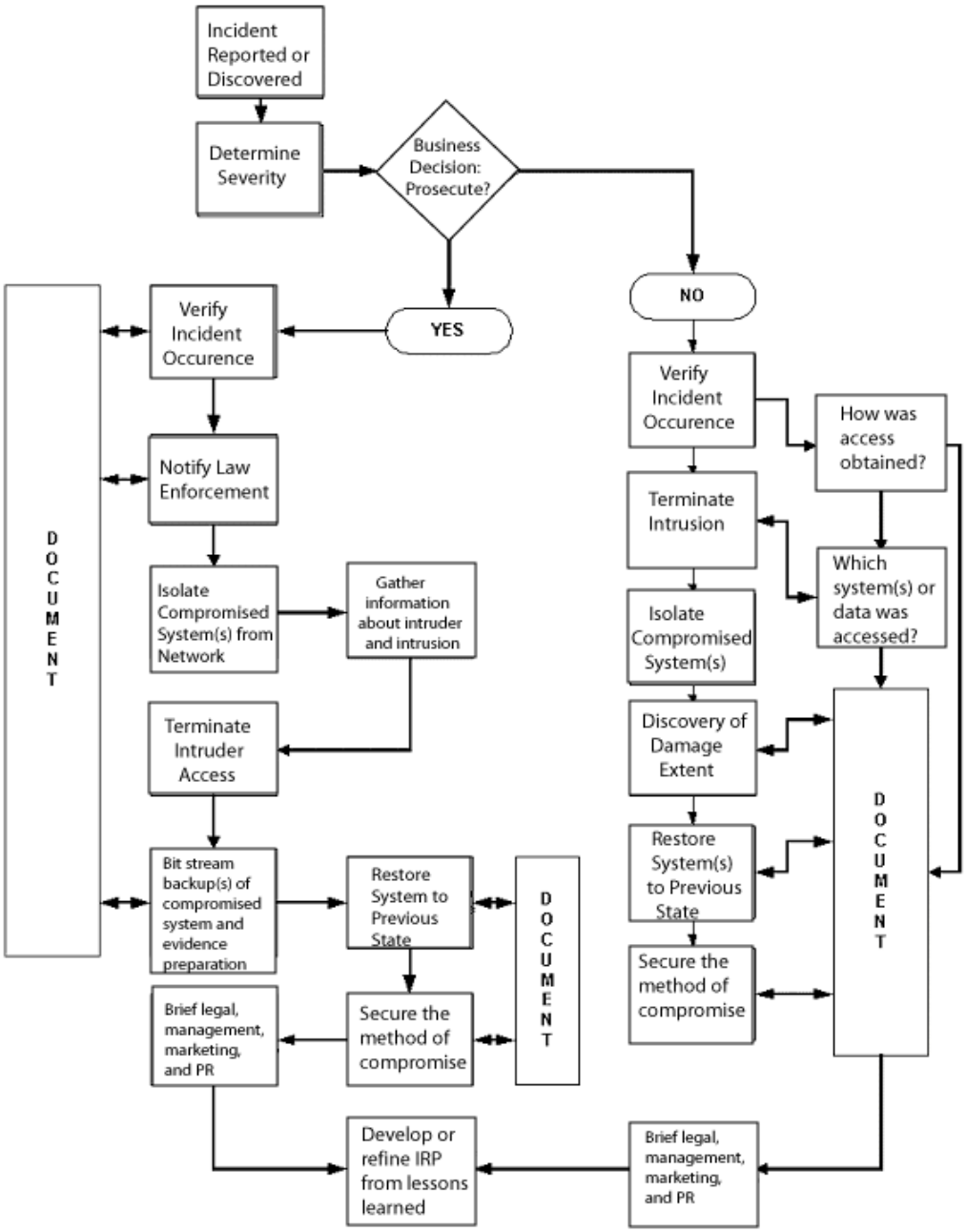


Figure 1 Sample IR flow plan utilizing all IR phases.

Alerting

The first phase in any IR cycle is alerting. This phase contains the process of learning and gathering information about a security incident, notifying the appropriate people, and following through on this notification. Make sure you establish from the onset what your notification schedule will be. Getting too many notifications is almost as bad as not being notified at all. Use the classification baseline you established previously to assist in determining alert frequency. These alerts are typically gathered or sent by several sources including intrusion detection systems (IDSs), firewalls, antivirus software, threats received via email, vendor newsletters, news and media, and so on. Your IRT can typically be notified in several ways such as phone, email, and duty phone/pager/fax; however, a good practice is to use a secure medium to communicate the incident after this initial notification. Get the word out as expediently as possible first to the right people, then communicate securely. In other words, it's generally acceptable to "yell" first and "whisper" thereafter. Ensure that this contact information is widely known, accurate at all times, and published to all who have legitimate need for it.

Investigation

The investigation phase begins with determining whether a suspected incident has validity, then establishing and classifying the severity of the incident, determining the quickest and most efficient way to counter the active attack, isolating the intruder, and documenting all steps. The IRT manager will typically have this responsibility in concert with other IRT members. Depending upon the severity of the intrusion, whether assets have been breached or compromised, and the company's written policy regarding security incidents, a decision is typically made at this stage to pursue prosecution of the intruder or deny access to the intruder and not prosecute. This decision is a business decision that is typically made at the highest executive level; therefore, it is essential that all feasible "what if" scenarios be worked out beforehand, much like the planning that goes into BCP. Many large companies make use of an automated ticketing system for tracking, incident logging, and communications.

If you decide to prosecute, make sure you have documented procedures for *everything*. All phases and aspects of the incident's entire investigation and specific timeline must be documented to the letter. The first thing a good defense attorney will do is to try to make it seem that your processes are in error and that you are mistaken in some way. Any room you can remove between the legal chasm of the actual facts and reasonable doubt will go a *long* way.

Responding

In the response phase, actions are typically based upon which decision was made in the investigation phase regarding prosecution of an incident. In a case of non-prosecution with no system compromise, applicable actions can be as simple as denying access to the system under attack or blocking specific or a range of IP addresses. If the prosecute option was chosen or there is a reasonable expectation that it will be chosen, evidence is collected, preserved, and documented in a forensic manner. If you don't have someone trained and experienced in computer forensics, you may choose to contract this step to someone qualified in this area.

Once evidence has been gathered and preserved, it is analyzed to determine the root cause of the incident, vulnerability(s) exploited, and vulnerability elimination/mitigation. Frequent assessments should be made during the course of attack to determine whether the incident is rising in severity, if any additional systems have been compromised, and so on.

There are some people who advocate the luring of an attacker into a honey pot environment after they have made the decision to prosecute an intrusion. The basic idea is that by luring the intruder into an environment away from valuable data, it will give more time to collect more evidence, information about the attacker, and potentially steer him or her away from more critical data. The honey pot can also become a very useful tool for learning and training purposes, but this concept is not something that will work for everyone. If you choose to create a honey pot environment, realize what you're getting yourself into. Think of it this way: If someone breaks into your house, would you lead him or her to the fake jewelry while the police are on their way? It's a safe bet you wouldn't. A honey pot requires care and feeding just like any other

critical system. If must be properly maintained, used, and monitored. In short order, it can become a gateway to critical systems with just one inadvertent slip up. Maintaining any sort of integrity once an intruder is beyond the perimeter is a daunting task, and even more so with a honey pot.

Recovering

The recovery phase begins when the response phase has been completed. These two phases are by no means mutually exclusive, as quite often some degree of overlap will exist. Such is especially true in large environments or when certain aspects of IR are contracted out. In this phase, all systems affected by the incident are returned to pre-incident configuration. In some cases, the affected systems may need to be completely rebuilt from tape backup, image, or scratch. If there is any doubt as to whether a system has been compromised, assume that it was—especially if you have a number of hosts "sporting" the same vulnerability located in the same area as the compromised system or device. After system restoration, correct the vulnerability and test the system. Any additional mitigation measures (such as additional firewall rules, device configuration, and so on) should be implemented, documented, and tested along with the compromised system.

Lessons Learned

The lessons learned phase is also commonly referred to as the "maintenance" phase. In this phase, the entire incident, from start to finish (with all documentation), is reviewed to determine which parts of the overall IRP worked correctly and effectively and where improvement can be made. When areas of deficiency are corrected, the IRP is updated to reflect these changes. These areas of deficiency are not just policy and response related. The lessons learned should also reflect any changes to system configuration (such as system or device hardening) and architectural changes that should be made (addition of another firewall, deployment of another IDS, and so on). This phase is also an excellent time to look at Service Level Agreements (SLAs—if any exist) to see whether they are effective or need to be adjusted.

Event Correlation and Aggregation

Located at the very heart of effective IR is an effective correlation and aggregation methodology. In smaller environments, this methodology is typically a manual and tedious affair of collecting information from systems and devices, weeding out false alerts, and analyzing the data considered to be valid after it has been though several iterations of parsing. Think of this methodology as putting information into a funnel. In the top of this funnel all logs, traps, events, and information is inserted, and out of the bottom, legitimate data emerges for analysis and action. How well this "funnel" works depends upon a few factors:

- The size of the funnel—The more information you allow in, the more you'll get out to have to run through analysis and act upon. Never collect more data than you could possibly stand to interpret within your specified reaction timeline. For this reason, if you are considering the idea of placing an IDS outside of your perimeter for analysis and statistics—such as seeing which attacks aren't making it through your firewall—plan to spend a great deal more time tearing down these events to get anything really useful out of them.
- The event correlation procedures or mechanisms used—As the saying goes, "garbage in, garbage out." If you are manually funneling information, there is a much greater chance of missing something. Correlation procedures or mechanisms that don't have the capability to automatically parse non-relevant information from the relevant will contribute to this problem.

If you have many events and devices to monitor, an event correlation engine will make your life tremendously easier. The cost of a good aggregation and correlation engine can be easily quantified into Return on Security Investment (ROSI) based upon the number of hours saved in manually processing events, assessing threat variation, and the decrease in both response time and your asset exposure window.

Proactive Vulnerability Scanning

A proactive vulnerability scanning plan can accomplish several things. First, it's a proven technique to reduce your active incident rate by finding vulnerabilities before the "bad guys" do. Second, it makes a good IR "fire drill" if done in a controlled manner. Notice that I said *controlled manner*. You should never, under any circumstances, point a scanner at a production system without having a plan and methodology in place associated with it. Many scanners and tools (both commercial and open source) can be *very* aggressive when they test systems for exploitable conditions. No one wants to inadvertently launch a DoS attack on their own systems.

Summary

In summary, an effective IRP is only as effective as the processes contained within it. These processes *must* reflect your business needs, structure, and budget. Technology has come a long way since the first IRP was put together, and products exist today that can accomplish nearly all of the mundane and often difficult-to-implement tasks in a very efficient manner. Through the elimination of redundant and repetitive tasks, the removal of human error from the equation, and the application of a holistic and repeatable approach, ROSI will no longer be a fictional entity to your organization—it becomes quite achievable.

###

About the Author, Ken Pfeil:

Ken Pfeil is Chief Security Officer of Capital IQ, based in New York. Ken's IT and Security experience spans over 18 years with companies such as Microsoft, Dell and Merrill Lynch. He is coauthor of "Hack Proofing Your Network, Second Edition", the upcoming "Steal this Network: How to Own the Box" (Syngrss), and a contributing author to "Web Services Security" and "The CISSP Study Guide". Ken holds a number of industry certifications, and participates as a Subject Matter Expert for CompTIA's Security+ certification. Ken is a management member of the CASPR (Commonly Accepted Security Practices and Recommendations) team, and is a member of ISSA's International Privacy Advisory Board. Ken is a member of the New York Electronic Crimes Task Force, IEEE, IETF, and ISSA. Ken is a frequent speaker at many security industry events around the world, including CyberSecurity 03, Infragard's National Security Conference and Gartner's Sector 5 Security Symposium.

About GuardedNet:

GuardedNet Inc. delivers advanced security management and incident response solutions. Its leading software solution, neuSECURE, centrally monitors, correlates and performs threat analysis in multi-vendor enterprise security environments. Its ability to correlate and analyze log data files from disparate machines in real-time enables security administrators to overcome log data overload and detect and respond to security breaches as they are occurring, rather than after the damage is done. neuSECURE has improved the security and the operational efficiency of numerous security operations centers (SOCs), including those at leading financial, transportation and data communications institutions. GuardedNet is a private company, headquartered in Atlanta, Georgia. For more information about GuardedNet, please visit www.guarded.net or contact us at sales@guarded.net or 888.599.8297.

For more information about how neuSECURE can enable centralized, real-time incident response in your organization, contact sales at 888-599-8297, sales@guarded.net or visit us on the Web at www.guarded.net.