



The Offensive Operations Model

Written By: Karsten Johansson

KSAJ Inc. - www.PENETRATIONTEST.com
163 Sterling Rd., Suite 9
Toronto, ON
M6R 2B2

(416) 732-5725
(866) KSAJ-INC

Version: 2.1 – April 2, 2004

0 Table of Contents

1	Licensing Introduction	3
2	Executive Summary	4
3	Why Do We Need an Offensive Operations Model?.....	5
3.1	The Scientific Method.....	5
3.2	Vulnerability Weight	6
3.3	Controls.....	7
3.4	Scalability	8
4	Offensive Operations Model.....	9
4.1	Model Summary.....	9
4.2	Understanding the Offensive Operations Model	10
4.3	Elements at Risk of Compromise	10
4.4	Characteristic Stages of an Attack	11
4.5	Categories of Control.....	12
4.6	Application of the Offensive Operations Model.....	13
5	Interpreting the Offensive Operations Model.....	14
5.1	Reconnaissance, Assessment and Strategy	14
5.2	Exploitation and Invasion	15
5.3	Maintaining Access.....	16
5.4	Operations	17
5.5	Conclusion	17
6	Using the Offensive Operations Model to Track Security Issues.....	17
7	Licensing.....	18

1 Licensing Introduction

This model was developed by Karsten Johansson under copyright ©1997 KSAJ Inc. It is provided to you under the Open Methodology License. A transcript of this license can be found in [Chapter 7 Licensing](#) of this document.

Note that if you choose to use this methodology, we're alright with that – we'd sure like to hear how you've implemented it. Email oom@penetrationtest.com with the title "Offensive Operations Model". If you choose to describe this model in your own documentation, give us credit.

2 Executive Summary

They say the best defense is a good offense.

If this is true, then one must question why information security is always in defensive mode:

- A virus attacks through employee email, so they patch against the virus.
- Something goes wrong with the database, so they patch the database
- Hacker tools are found on the web server, so they remove them, and patch the web server
- Another virus strikes, and they patch again

By acting defensively all the time, corporations find that in terms of security, the bad guy is exactly one step ahead at all times. Inevitably, this is a costly cycle that never really ends. Constant patching and updating has not solved the security problem, and as long as this cycle is designed into our practices, it will always be difficult to fully realize any Return on Security (ROSI) Investment.

Security maintenance doesn't have to be this way.

The Offensive Operations Model provides a framework for planners, developers and testers to look at security using the same concepts and processes one uses when attacking. By doing so, most risks can be mitigated from the onset of a project.

Typically, developing and testing rely on completely different models. This makes testing the effectiveness of the development to be unreasonably complicated, and makes it impossible to develop realistic security metrics. The Offensive Operations Model is scalable, so its concepts span the initial planning all the way through the testing and auditing.

This document describes the model, and provides examples of how it can be implemented at any and all stages of the information security lifecycle - from planning, to development, to deployment, to auditing and testing.

3 Why Do We Need an Offensive Operations Model?

Security is a moving target. Security professionals are all too aware of this: the more we patch our systems, the more unpatched holes we discover. Much of the industry treats Information Security as if it is something new. In reality, all the attacks we see now are variations of attacks that have been around for centuries.

For example:

Viruses	Viruses were “invented” in the 70’s and have become a widespread problem since the mid-80’s. The probable first mention of computer viruses appears in David Gerrold’s “When H.A.R.L.I.E. Was One” in 1972 (reprinted with updates in 1988).
Identity Theft	Identity Theft is probably the world’s other oldest profession, and was Jacob’s modus operandi when he disguised himself as Esau to steal the family birthright from their father Isaac in the Bible book of Genesis.
Distributed Attacks	Nothing new. Sun-Tzu described different attack methodologies in great detail 2400 years ago.
Encryption	Julius Caesar encrypted his messages by shifting each letter of a word by 3.

Instead of treating information attacks as something new, the Offensive Operations Model focuses on each stage of an attack, and what controls would mitigate the associated risks. It is a proactive model, versus the reactive stance most security organizations are currently taking when they rely on the Scientific Method as their only defense.

3.1 The Scientific Method

Other than by sheer accident and surprise, most discoveries occur as the direct result of methodic cause-and-effect evaluations of potential solutions in order to solve a known problem. The most notable description of this process is known as the Traditional Scientific Method, by which the path to discovery is a process involving the five following activities:

1. Clearly define the question or problem
2. Gather any available data related to it
3. Form a hypothesis
4. Experiment and document the results
5. Accept, modify or reject the hypothesis

One could argue that penetration testing follows these steps quite literally:

Defining the question in this case is fairly straight-forward: How could a dedicated attacker threaten the corporate IT investment? Gathering of relevant data occurs on many levels,

from reviewing what is on the network, to researching what ways the software is known to be vulnerable. The hypothesis is simply “an attacker can do X to gain access to Y”, which is then proved or disproved by attempting the attack and observing whether or not it was successful. If successful, the tester accepts the hypothesis, and records the observations in a report. If not successful, the tester changes their approach, or rejects the hypothesis, and thus can claim that the system is not vulnerable to that particular attack.

The object of a model is to produce consistent behaviors and actions, and in terms of penetration testing, this model does at first seem to be a good fit. In fact, it is the model used by most penetration testers, whether they realize it or not.

A problem with using this type of model for testing security metrics is that it is not sufficiently scaleable. If one is to believe that security begins at the policy level, then the models used for writing those policies should be the same as those used for testing their implementation. The Traditional Scientific Method works well for understanding and testing specific technical problems, but does not scale very well to other elements of security.

A good model recognizes the conceptual and theoretical phenomena surrounding a particular discipline. It seems valid to say then, that an appropriate model would be scaleable enough to both define and test security at the technical, managerial and operational levels in order to weight the issues and costs appropriately.

The time between vulnerability discovery and vulnerability exploitation is getting shorter all the time. Security practices need to be streamlined and efficient while maintaining a high degree of thoroughness. In order to demonstrate any actual level of Return on Security Investment (ROI), measurable results are required.

The Traditional Scientific Method is too basic (consisting of reconnaissance, attack, more reconnaissance, more attack, etc.), as it lacks a standardized way to interpret results and does not provide a reliable mechanism for determining the actual severity of what was discovered. In most cases, such a methodology is better for testing the tester’s ability than for testing actual security.

3.2 Vulnerability Weight

Likewise the Traditional Scientific Method does not provide a framework for developing or measuring the non-technical aspects of security. Vulnerabilities range from being purely hypothetical to being quite serious exposures. Many hypothetical exposures become very real once the associated vulnerability becomes widely known. Not so long ago, the method for cracking WEP keys was purely hypothetical. Now anyone with a wireless laptop can break the strongest WEP keys in a matter of hours.

As well, not all technical vulnerabilities are due to software and hardware bugs. This leads the tester to apply a sometimes arbitrary weighting mechanism in order to prioritize each issue. Most people call this guesswork, and security is not about guesswork.

The weighting should be accurate and stringent enough for the corporation, not an attacker, to affect any changes to it. An example we have seen in the industry, but have pretty much abandoned for this reason, is the ECHO ETUN weighting system. The ECHO rating specifies the issue as being Extreme, a Concern, a Housekeeping issue, or OK. The ETUN rating specifies the level of ability an attacker needs to exploit the vulnerability - Expert, Technical, User or Naïve.

Without much thought, this seems a valid way of measuring the level of risk. But this oversimplification has many shortcomings. For example, a relatively serious vulnerability that would take an Expert to exploit might be rated CE (Concern / Expert). Immediately thereafter, such an expert could create and publicly release a program, such as a worm, to actively search out and exploit that vulnerability. The security industry has seen this time and time again. Suddenly that CE rating is no longer valid because a Naïve user might simply “click on the pretty icon”, and the rating suddenly changes to an Extreme vulnerability that a Naïve person may exploit (EN). An entity external to the corporation affected this change to the rating, and therefore the weighting mechanism is clearly defective.

With the immediacy of obtaining exploit source codes from the web, or the rapid spread of worms, security professionals are best advised to assume all vulnerabilities are simple enough for a moderately technical to completely naïve person to exploit. Indeed Mafiaboy, who brought CNN, Yahoo and a few others to their knees, was no genius.

When each step of an attack is taken into account, it is then possible to weigh the actual severity of vulnerabilities by their particular exposure characteristics. For example, a given vulnerability may be extreme, but only exploitable in the final stages of an attack. By implementing good security controls to hamper an attacker’s ability to reach that stage of the attack, much of the risk is mitigated regardless of presumed skill level requirements or hacker prowess. This will become clearer once the model itself is understood.

3.3 Controls

Like most reactive security models, the Traditional Scientific Method is primarily concerned with the knowledge of specific tools and techniques. It does not take controls into account at all, even though security professionals claim it to be the primary domain of security policies, standards and procedures. This leaves those in charge with security no recourse but to keep patching holes as they are found, instead of designing their resources to be resilient to these problems in the first place.

The Offensive Operations Model was developed by KSAJ Inc. and published on www.PENETRATIONTEST.com in order to present a solid security framework that applies to both the development and the testing of security mechanisms at any level. This sounds nearly impossible, and perhaps quite complex, but the model is purposely developed around some very well understood security principles, making it a fairly simple model to understand and implement.

Experience proves time and time again that there is no such thing as a 100% secure system. Acknowledging the fact that attacks are not particularly spontaneous, the Offensive Operations Model avoids looking at security from a purely “technical exploit” perspective. Instead, the focus is on security controls that mitigate the risks to data and information inherent to each level of attack. Instead of trying to ensure that every possible bug has been patched, and blaming security lapses on delayed or missing patches, it is much more beneficial to ensure that controls are in place to reduce the probability that a dedicated attacker would be successful in the first place. In many ways, this is not too different than the philosophy of Defense in Depth, and perhaps could be seen as an abstracted view of it.

3.4 Scalability

The Offensive Operations Model provides the level of scalability we need - policies, architectures, and anything else related to security can not only be tested, but even developed in a way that all levels of attack are thwarted by design. For example, since reconnaissance is the first stage of an attack, a developer can significantly strengthen the security posture by implementing controls to reduce the level reconnaissance exposure. Likewise, once implemented, the tester now has actual metrics by which to measure the level of security. Herein lies the Return on Security Investment (ROSI).

Patching is another area of security we've already mentioned where the Traditional Scientific Method isn't a very good fit after all. Because of that, administrators spend about 80% of their time and effort patching and reconfiguring every day. Interestingly, 80% of the patches that are released do not increase the security of the systems they get installed on. The Offensive Operations Model provides a way to determine if a patch is even relevant for each system that may contain a known bug, and a way to prioritize the activities addressing the real risks.

We are no longer restricted to reviewing purely technical vulnerabilities from a diagnostic standpoint. Use of the Offensive Operations Model allows the planners, developers, and the testers to understand the security implications of every area of the network where data is transported, processed or stored, and develop their architecture, policies, and procedures around them.

The next part of this document describes the Offensive Operations Model in detail - the metrics, how they influence each other, and how they are applied in a thorough and conscientious security program. As well, examples are provided to demonstrate the scalability of the model.



4 Offensive Operations Model

The term “security” means different things to different people in an organization. To upper management, security involves financial risk. To the receptionist, security involves filtering external access and communication, in person or via the telephone, fax or mail, to staff members within the organization. To security guards, security is about preventing unauthorized physical access to the premises. To network administrators, security involves firewalls and passwords.

This is because security covers such a wide range of topics at all levels of the organization. The Offensive Operations Model was designed to overcome differences in implementation – the model used to develop the corporate architecture should also be the model by which it is audited. The model by which physical security is defined should also be the model by which electronic access is defined. The security wheel should not be reinvented for each level of the organization or for each security requirement.

This model does not describe specific interviewing and assessment techniques, or specialized software tools, but does provide the framework for determining their effectiveness in the capacity that they are or could be used. As well, it will uncover where tools and techniques should be employed.

Likewise, security is critical – it is important to document all tests and discoveries and possibilities, no matter how minimal they appear. A seemingly uninteresting vulnerability might be a very big deal if used in tandem with other vulnerabilities. Above all, the Offensive Operations Model excels at finding these vulnerabilities before they can become a security issue.

This section provides a summary of the Offensive Operations Model, and then describes its individual parts.

4.1 Model Summary

The Offensive Operations Model is based on the very stages a dedicated or professional attacker takes to exploit and covertly operate through a system or resource they wish to compromise. The type of attack is irrelevant – robbing a bank, cheating credit card holders into disclosing their confidential information, penetrating corporate network resources – all systematic attacks follow a similar process.

In order to conduct offensive operations, a professional thoroughly investigates their target, carefully researching areas where they may be vulnerable. Discoveries that are potentially useful in an attack are noted, and then reviewed to develop an attack plan that is most likely to succeed with the least chance of being detected or contained. They then attempt to exploit these potential vulnerabilities in a similarly structured manner while consistently monitoring the effect to determine their progress.

All this has occurred well before the operations has even begun.

Once an attacker has determined a successful method of exploiting the target, they may implement a backdoor method, or some other way to make re-entry possible. At this point, the target is completely compromised, and the offensive operations can commence.

This type of attacker tends to be far more thorough than a hobbyist hacker who generally downloads hacker scripts as they become available. It is due to this thoroughness that they have a much higher success rate and often enough, do not get caught in the act.

Once each Stage of an Attack is understood, it is possible to quantify the level of risk, and then apply appropriate controls to mitigate those risks.

4.2 Understanding the Offensive Operations Model

Risk can be measured by understanding the interplay between the stages of a successful attack, the types of risk to information or data, and the types of security control. The Offensive Operations Model breaks these aspects down into a few basic rules:

1. Risk is the probability that a Vulnerability will be exploited
2. Wherever information or data is processed, stored, or transported, there is a risk of exploitation of its Confidentiality, Integrity, Availability and Accountability
3. Risks against data or information increase in severity at each stage of an attack
4. The Elements at Risk of Compromise, taken in context with each Stage of an Attack determines the type of Controls that need to be applied

These rules can be applied in several ways:

- Appropriate Security Controls can be designed into a network by what is at Risk throughout each Stage of an Attack.
- Security Controls can be tested by conducting activities associated with each Stage of an Attack and determining the risk that vulnerabilities can be exploited.
- The level of Risk is directly proportionate to the strength of the security Controls at each Stage of an Attack.
- The Bad Guy™ succeeds through the Stages of Attack by discovering weak or missing Controls and exploiting the inherent Risks.

4.3 Elements at Risk of Compromise

The first order of business is to understand Risk. Risk is the possibility of suffering harm or loss. Companies describe their qualified or quantified risks in their business plans, SEC filings, security policies, etc. Security is no different. In fact, a security breach is simply an exploited risk.

Following are the four elements at Risk of compromise, with examples of datasets that could be affected:

1. **Confidentiality** – Closely held information, such as passwords, algorithms, personal information, company secrets, crypto-keys, etc.
2. **Integrity** – adherence to a strict value or form, such as for accounting tables, data streams, original system binaries, methodologies, etc.

3. **Availability** – accessibility of a resource when it is required or expected, such as network bandwidth, web or email server, logs, backups, etc.
4. **Accountability** – ownership of data or processes, such as a user account, network traffic from an application, email, administrative duties, etc.

4.4 Characteristic Stages of an Attack

An attack is the act of exploiting or attempting to exploit risks. All successful intrusions share the following five characteristic phases:

1. **Reconnaissance**
 - a. Scoping out a target
 - b. Inspection or Exploration
 - c. Preliminary examination or survey
 - d. Preparatory expedition
 - e. Eyeballing
2. **Assessment and Strategy**
 - a. Classification with respect to value of collected data
 - b. Creation of an elaborate and systematic plan of action
 - c. Further actions that need to be executed as soon as possible
3. **Exploitation and Invasion**
 - a. Employing the greatest possible advantage over a target
 - b. Take advantage of a security hole
 - c. Attack to gain entry
4. **Access Maintenance**
 - a. Back door
 - b. Cracked user accounts
 - c. Unused or default accounts
 - d. Facilitate simplified re-entry
 - e. This stage may be skipped if it is a one-time operation

5. Operations

- a. Final goal of the attack
- b. Web page defacement
- c. Data or identity theft
- d. Theft of services

The main Risk in the Reconnaissance Stage of an Attack is against Confidentiality. An attacker is typically very cautious not to exploit any other Risk elements until enough information is gathered, avoiding activities that may set off an alarm.

Reconnaissance can go undetected for considerable lengths of time and the Assessment and Strategy stage is often completely undetectable, as it is usually done without contact with the target. This means that often security controls are most apparent at the Exploitation and Invasion stage.

It is important to note that each phase is conducted in such a way as to ease the way for the next step, and lower the chance of being observed by the target. In practice, more time spent on one step ensures better results in the subsequent steps. The attacker's skill at reconnaissance can arguably make or break an intrusion attempt.

Each stage will often present conditions that prompt the attacker to revisit a previous Stage of Attack. For example, if the attacker realizes not enough information was collected to complete an Assessment and create a thorough Strategy, they will conduct more Reconnaissance on the target. If insufficient access is available to conduct Operations, the attacker will attempt further Access Maintenance activities until they have obtained sufficient access.

4.5 Categories of Control

Strictly speaking, a control is a mechanism to reduce risk. This may entail blocking data flow to outside networks, ensuring data integrity, or maintaining its accessibility. Controls also provide functions to notify when an attempt has been made to circumvent allowable access, and an audit trail to accurately document differences. Most controls are focused on a limited number of threats or vulnerabilities, and singularly can be defeated. Because of this, a robust suite of controls is necessary to mitigate risk.

There are 6 categories of Control:

1. **Prevention** – Hindering an attack, or imposing an obstacle
2. **Detection** – The perception that something has occurred or that some state exists
3. **Containment** – Controlling the expansion or influence of an action or state
4. **Eradication** – Elimination of a problem, removal of evidence from a crime scene
5. **Recovery** – Return to normal condition or original state
6. **Follow-up** – Review and evaluation of activities and solutions – lessons learned

Much like the Stages of an Attack, it is important to note that each Control helps and strengthens the subsequent controls. Prevention is the first line of defense. When Prevention fails, we hope the attack is Detected. Containment helps ensure that other systems are not affected, or are affected as minimally as possible during the attack. This makes simplifies Eradication of the attacker, which allows for an efficient Recovery. Once recovered, Follow-up allows for tweaking of the controls to mitigate risk of an attack reoccurrence.

In practice, controls are very cyclical. Follow-up information should be used to strengthen Prevention controls.

4.6 Application of the Offensive Operations Model

Given the 4 types of exploits and the 5 characteristic phases of a successful intrusion, policies, procedures and assessments can be developed around the 6 categories of control. For each stage of an attack, at least one of the risks of exploitation are used, and often in several possible ways. In other words, for each stage of the attack, each of the 6 categories of controls are tested and weighed against their potential associated risks.



5 Interpreting the Offensive Operations Model

The Offensive Operations Model is highly scaleable, and therefore has many applications for those who deal with intrusions on virtually any level. This section provides examples of some of the risks and applications of controls to mitigate them at each stage of an attack.

Assessing the security of any system or environment requires a deep understanding of what is at risk through each step in the case of an attack. A Security Assessment is then a matter of questioning the controls around each given risk.

In the following series, each stage of an attack is listed, along with each control category. Each of these is provided with examples to demonstrate assessment of the control. Throughout the procedure – whether it is policy development, architecture design, application deployment, penetration testing, or security auditing – the spirit of the Offensive Operations Model remains consistent, and all combinations should be considered.

5.1 Reconnaissance, Assessment and Strategy

Reconnaissance, or Recon, is the act of scoping out a target. This information gathering stage is the most important step an attacker takes, and all key information they can gather is considered for its relevance to the attack.

The Assessment and Strategy stage is the sorting of the gathered data to piece together a relatively complete picture of the target's potential weaknesses. Although distinct, these two Stages are so inter-related that the controls against the primary Stage directly affect the latter.

In Reconnaissance, knowledge is gained either through Inference, or through Credentials. Inference is the deriving of conclusions through reasoning, whereas a credential is known or demonstrable fact. Both must be considered:

- A classified ad seeking a person to research and implement an Intrusion Detection System infers that at least some part of the network is not protected by IDS.
- Knowing that John Smith's email address is `jsmith@hackme.corp`, one can infer that his workmate Jane Doe's email address is `jdoe@hackme.corp`.
- Capturing a server banner that displays the version and patch number and list of plug-ins, and verifying that information is reflected in error messages, is knowledge through credentials
- Scanning a network to map out what systems are responsive is knowledge through credentials.

When assessing controls around the Reconnaissance and Assessment & Strategy stages of an attack, each of the 6 areas of control needs to be identified. Example questions are given, although in real-case scenarios, questions will be more directed to what is being assessed.

Reconnaissance information is gathered using both passive and active techniques. For example, searching for information about the target on Google is passive, since no actual

connection is made to the target system. A port scan however, is active, since a connection is made to the target, and could potentially be witnessed by someone else monitoring the target.

Following is a list of the Controls with examples of how they apply to the Risks inherent to Reconnaissance:

1. **Prevention** - Do server banners provide overly detailed information about the system or network? Do login scripts behave identically for failed usernames as with existing ones? Is directory browsing through http disabled?
2. **Detection** - Is logging effective?
3. **Containment** - Could information obtained from the server provide information about neighboring systems and networks?
4. **Eradication** - Is a method in place to remove, block, or change data that may be used to create more specific attacks against the server?
5. **Recovery** - Can system functions "recover" old data that was previously removed to prevent reconnaissance?
6. **Follow-up** - Are post-mortem reports generated when there is potential evidence of data-mining? Is a record kept of all dates, times, IP addresses and suspected network-mapping activity for reference if a penetration or other crime is committed?

5.2 Exploitation and Invasion

Once a hacker has gathered enough information and has pieced together a reasonable amount of information about the network or system they are attacking, and have devised an initial plan of attack, it is then possible to begin the Exploitation and Invasion stage. At this point, the hacker uses the gathered knowledge and attempts to access the server through the channels that were found open.

In this Stage of Attack, all of the Risk elements must be considered since the Confidentiality, Integrity, Availability and Accountability may be targeted or affected. These Risks may be exploited purposefully, or might be a result of mistakes made by the attacker.

When assessing the controls around Exploitation and Invasion, the following types of questions can be asked:

1. **Prevention** - Have appropriate security patches been applied? How are buffer-overflows prevented? Is the access control sufficiently strong? Could the system play host to a Distributed Denial of Service attack?
2. **Detection** - Would a successful penetration be detected? Is a method in place to notify individuals who are in authority to react to an incident?
3. **Containment** - Can penetration of a single service cause compromise in other servers? Can a single service be used to gain control of the entire system?

4. **Eradication** - Do controls allow for scalability when a compromise dictates change? Do controls protect potential forensic data?
5. **Recovery** - Can the system be brought back online in a short period of time following a worst-case intrusion? Can backups be relied upon to not re-introduce the vulnerability?
6. **Follow-up** - Are detailed reports of an incident and its mitigation generated by the system?

5.3 Maintaining Access

Once a hacker has penetrated the network (or if the hack is an inside job) steps are usually taken to make future accesses easier to conduct. This may involve installing a back-door program, but sometimes may be something as simple as setting up home base under a seldom-used account name or identifying a misconfigured user account with suitable permissions to use to regain entry.

This stage of the attack directly affects Integrity and Accountability, and may result in residual Risk to Confidentiality and Integrity.

When assessing controls that limit hackers from increasing their level of access to improve re-entry, the following types of questions can be asked:

1. **Prevention** - Is effective change detection software installed and enabled? Is the system administrator alerted when access levels and permissions have changed?
2. **Detection** - What features are running on the system to detect back-door programs, or critical file-system changes? Are port-scans conducted to detect possible back-door network services?
3. **Containment** - Could a backdoor installed on the system be leveraged to attack another system?
4. **Eradication** - Are features in place for reassigning access levels or permissions if they have been changed? If a back door such as Back Orifice is discovered, is the system immediately taken offline and rebuilt from a known-good image?
5. **Recovery** - Is sufficient change control in place to avoid accidentally re-introducing a backed-up version of a back door mechanism? Do these features inhibit other system functions negatively?
6. **Follow-up** - Are detailed reports of suspicious file permissions generated? Are logs sufficiently detailed to investigate the source of back doors or Trojan files and prevent similar compromise?

5.4 Operations

This is the most dangerous part of a penetration - the attacker has all the access required to carry out their agenda. If it is a spy operation, data could be sent to a remote collection repository. If it is a system-mapping reconnaissance mission, existing levels of access may be used to compromise more systems on the network.

In this stage, as with Exploitation and Invasion, all Risk elements must be considered.

When assessing controls that limit illicit operations on systems and the network, the following types of questions can be asked:

1. **Prevention** - Is mysterious executable code discovered on a system disabled and quarantined?
2. **Detection** - Is an alarm mechanism in place when potentially harmful operations are detected? Is someone with enough authority to investigate and react appropriately notified of a potential operation?
3. **Containment** - Is the system architecture robust enough to limit an attacker to a single environment? Does manipulation of a single target affect unrelated functions? Is a mechanism in place to minimize and control the extent of damage?
4. **Eradication** - When operations are detected, is a mechanism in place to end the activity and deal with it appropriately?
5. **Recovery** - Can a system be coerced into accidentally recovering a back-door mechanism or other malicious binary file?
6. **Follow-up** - Is a mechanism in place to follow an incident from the moment of suspicion to the point that the case is considered closed?

5.5 Conclusion

Use of the Offensive Operations Methodology can uncover security flaws in any feature, configuration or trust relationships, whether they be technical, architectural, design or policy related. Attacks are based on the existence or apparent existence of these weaknesses. A gap in any one of the areas described above is a potential security vulnerability.

6 Using the Offensive Operations Model to Track Security Issues

This section will be included in the next version of the Offensive Operations Model whitepaper. Official updates to this document are uploaded to <http://www.penetrationtest.com>, in the Whitepaper section. The author is unable to ensure versions available elsewhere are up-to-date, Comments and requests should be sent to oom@penetrationtest.com with the subject line of "Feedback", "Request" or "Offensive Operations Model". All other emails to oom@penetrationtest.com are sent to /dev/null, and thereby automatically deleted. I.e.: do not include this address in any mailing list. Anything resembling spam or Unsolicited Bulk Email (UCE) will not be received or acknowledged.

The purpose of this section, once it has been included, is to provide concepts and examples for using the Offensive Operations Model to conduct, document and track corporate risk analysis, security policy, network design, audits, and penetration testing initiatives. Sample documentation templates will be provided as well.

7 Licensing

This document was written and provided to you in the spirit described in the Open Methodology License. A copy of the following license may be found at <http://www.isecom.org/oml.shtml>

OPEN METHODOLOGY LICENSE (OML)

Copyright (C) 2000 - 2003 Institute for Security and Open Methodologies (ISECOM).

PREAMBLE

A methodology is a tool that details WHO, WHAT, WHICH, and WHEN. A methodology is intellectual capital that is often protected strongly by commercial institutions. Open methodologies are community activities which bring all ideas into one documented piece of intellectual property which is freely available to everyone.

With respect the GNU General Public License (GPL), this license is similar with the exception for the right for software developers to include the open methodologies which are under this license in commercial software. This makes this license incompatible with the GPL.

The main concern this license covers for open methodology developers is that they will receive proper credit for contribution and development as well as reserving the right to allow only free publication and distribution where the open methodology is not used in any commercially printed material of which any monies are derived from whether in publication or distribution.

Special considerations to the Free Software Foundation and the GNU General Public License for legal concepts and wording.

TERMS AND CONDITIONS

1. The license applies to any methodology or other intellectual tool (i.e. matrix, checklist, etc.) which contains a notice placed by the copyright holder saying it is protected under the terms of this Open Methodology License.
2. The Methodology refers to any such methodology or intellectual tool or any such work based on the Methodology. A "work based on the Methodology" means either the Methodology or any derivative work by copyright law which applies to a work containing the Methodology or a portion of it, either verbatim or with modifications and/or translated into another language.
3. All persons may copy and distribute verbatim copies of the Methodology as are received, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and creator or creators of the Methodology; keep intact all the notices that refer to this License and to the absence of any warranty; give any other recipients of the Methodology a copy of this License along with the Methodology, and the location as to where they can receive an original copy of the Methodology from the copyright holder.
4. No persons may sell this Methodology, charge for the distribution of this Methodology, or any medium of which this Methodology is apart of without explicit consent from the copyright holder.
5. All persons may include this Methodology in part or in whole in commercial service offerings, private or internal (non-commercial) use, or for educational purposes without explicit consent from the copyright holder providing the service offerings or personal or internal use comply with points 3 and 4 of this License.
6. No persons may modify or change this Methodology for republication without explicit consent from the copyright holder.

7. All persons may utilize the Methodology or any portion of it to create or enhance commercial or free software, and copy and distribute such software under any terms, provided that they also meet all of these conditions:

- a) Points 3, 4, 5, and 6 of this License are strictly adhered to.
- b) Any reduction to or incomplete usage of the Methodology in the software must strictly and explicitly state what parts of the Methodology were utilized in the software and which parts were not.
- c) When the software is run, all software using the Methodology must either cause the software, when started running, to print or display an announcement of use of the Methodology including an appropriate copyright notice and a notice of warranty how to view a copy of this License or make clear provisions in another form such as in documentation or delivered open source code.

8. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on any person (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If said person cannot satisfy simultaneously his obligations under this License and any other pertinent obligations, then as a consequence said person may not use, copy, modify, or distribute the Methodology at all. If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

9. If the distribution and/or use of the Methodology are restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

10. The Institute for Security and Open Methodologies may publish revised and/or new versions of the Open Methodology License. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

NO WARRANTY

11. BECAUSE THE METHODOLOGY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE METHODOLOGY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE METHODOLOGY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE IN USE OF THE METHODOLOGY IS WITH YOU. SHOULD THE METHODOLOGY PROVE INCOMPLETE OR INCOMPATIBLE YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY USE AND/OR REDISTRIBUTE THE METHODOLOGY UNMODIFIED AS PERMITTED HEREIN, BE LIABLE TO ANY PERSONS FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE METHODOLOGY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY ANY PERSONS OR THIRD PARTIES OR A FAILURE OF THE METHODOLOGY TO OPERATE WITH ANY OTHER METHODOLOGIES), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END.