

# Statistical-Based Intrusion Detection

*Jamil Farshchi* 2003-04-16

On January 24, 2003, the [W32.SQLExp.Worm](#) (later named Slammer/Sapphire) was released into the wild. This worm exploited a stack-based buffer overflow vulnerability in Microsoft's SQL Server 2000 software (including MSDE 2000). While vulnerabilities affecting Microsoft products are nothing new, the speed at which this worm propagated was extremely novel - scary in fact. The worm was released and within ten minutes it had compromised 90% of all vulnerable systems worldwide. Before this incident, worms of this type were merely theoretical, given serious consideration primarily in the academia.

It takes even the fastest vendors hours or days to produce a signature for rule-based intrusion detection (RBID) systems. In the case of this worm however, a vulnerable network would be compromised in a matter of seconds, much too quickly for even the most diligently updated RBID system. So what is the solution to a worm that doubles its infection rate every 8.5 seconds? The answer may lie in a lesser-known intrusion detection method called statistical-based (also referred to as behavior-based) anomaly detection.

## Intrusion Detection the Traditional Way

The most common intrusion detection systems are [rule based](#) (also referred to as signature based). RBID systems use an attack "signature" to identify a potential attack and subsequently alert on the suspect network traffic. Signatures can be made to look for anything from a port number in the packet header to a specific byte sequence in the payload of a series of packets. Once a signature has been developed and implemented, it is usually quite effective at detecting the said network activity.

The primary shortcoming of the RBID approach is the development and implementation of the signatures. Because RBID systems rely on signatures to detect malicious traffic, without the signatures, the effectiveness of an RBID system is significantly reduced. Unfortunately, rule development on even a moderately complex attack is not trivial. Development takes time. First, an attack must be detected, recorded, and analyzed. Second, the signature must be created, and finally, the new signature must be distributed. Once the signature is distributed, IDS operators still have to implement the new rule for the IDS (and sometimes this is no small feat). These steps take time and, as discussed earlier, time is something rapidly spreading worms such as W32.SQLExp do not bestow upon IDS administrators. Even if all the steps are

completed and implemented at breakneck speed, this fundamental flaw remains: if the attack is new and there is no signature, a RBID system may fail to alert on the activity.

## **Intrusion Detection the Statistical Way**

Statistical-based systems (SBIDs) take a different approach to intrusion detection. The concept of the SBID system is simple: it determines "normal" network activity and then all traffic that falls outside the scope of normal is flagged as anomalous (not normal). SBID systems attempt to learn network traffic patterns on a particular network. This process of traffic analysis continues as long as the SBID system is active, so, assuming network traffic patterns remain constant, the longer the system is on the network, the more accurate it becomes. By analyzing network traffic and processing the information with complex statistical algorithms, SBID systems look for anomalies in the established normal network traffic patterns. All packets are given an anomaly score (indicating the degree of irregularity for the specific event) and if the anomaly score is higher than a certain threshold, the IDS will generate an alert. The key to any SBID system is its ability to learn and distinguish normal from anomalous network activity.

### **The Basic SBID System**

Suppose you own a dog (Spot) and every day you come home from work and feed it. Day in and day out, you come home to an empty food bowl and you promptly fill it up for your canine companion. One day you come home and the food bowl is only half empty. Depending on your particular situation, this event could be very curious and warrant investigation, or it may be something that you do not care about. Now think if you came home one day and the food bowl is completely full. In this extreme case, it would be reasonable to assume that something is awry. Spot may be sick, Spot may have run away, or maybe he just wasn't hungry. Needless to say, due to the level of abnormality of the situation, further investigation would be necessary.

This is how SBID systems are designed to work. The system learns what is normal by observing an activity over a period of time (just like coming home to an empty food bowl each day). When an unknown or rare (anomalous) event occurs, the SBID observes it and subsequently generates an alert. (Just like coming home, noticing a full bowl of dog food and subsequently thinking something may be wrong with Spot.) The difference between an alert and a non-alert is based on the anomaly threshold. (This is similar to the instance when you come home and the bowl of dog food is half-full based on your situation, you determine if the event is worthy of further analysis.) If the anomaly threshold is high, minor anomalies are not noteworthy (you

don't really care if there are changes to the amount of food in the bowl). If the threshold is low, most of the anomalies are a cause for investigation (you will check on your dog and look into the situation if the bowl is half full).

## **The Good**

SBID systems have a number of advantages. The behavior-based analysis of traffic patterns and subsequent notification of anomalous activity can be critical in today's world of constant cyber threats. Because no prior knowledge of an attack is required, as is generally the case with rule-based IDS, SBID systems can detect "0 day" (extremely new) attacks. That alone is worth the price of admission, as traditional RBID systems are generally much less likely to detect 0-day attacks.

If a network is attacked by a previously unseen worm, virus, or Denial of Service (DOS) attack, a SBID system is likely to alert based on the presence of the unusual activity. SBID systems are also capable of detecting "low and slow" attacks. These attacks (or portscans), usually performed by skilled intruders, are characterized by their lengthy duration (possibly months at a time), precision, and methodical execution. Usually these attacks are intended to enumerate the network or gather information about a specific system. It doesn't matter how "low and slow" that attack is though, in most cases, even if the attacker sends a mere packet a month (or less), the SBID system will note that it is anomalous traffic and alert on the event.

Another major benefit of a SBID system is that it is potentially easier to maintain than an RBID system. There is no need to update signatures because the system doesn't rely on specific attacks or conditions. You may say: "A self-updating, "0-day" attack catching, portscan notifying, behavior based IDS? This must be the panacea of intrusion detection." Well, not exactly.

## **The Bad**

SBID systems are wonderful for what they can potentially accomplish, but there are a few fairly significant shortfalls in the technology. A SBID system's usefulness relies completely upon its ability to learn the regular patterns of network traffic for a given network segment. Theoretically, this principle is sound, but not necessarily in practice.

The truth is that most networks are extremely diverse in terms of protocols, services, and

usage times. In many cases, the only thing "normal" about a network is the fact that it is constantly changing. SBID systems also suffer from the ability to be "taught" by intruders. For example, an attacker could use a program like Nmap and send numerous SYN-scans at the network. Over time that activity would be deemed normal by the SBID system. The SBID system learns what is "normal" for a given network so if Web traffic, for instance, is the only activity on a network, anything that falls outside of that scope will be alerted on. On the flip side, if flood pings are commonplace on a given network, then a SBID would allow the activity because it cannot determine "good" traffic from "bad", only normal from anomalous (This is analogous to a situation where the "check engine" light in your car turns on. Initially it is a concern, but if it continues to stay lit with no noticeable problems to the car, the light becomes more and more ignored. Eventually, the illuminated warning light becomes "normal" to you).

Another issue with statistical-based intrusion detection is the implementation of such a system. Unlike a RBID system, a SBID system will not have an immediate effect when it is connected to the network - it cannot hit the ground running. Even if the SBID system is on a relatively consistent network, the learning process takes days or weeks to become accurate and effective. SBID systems require that operators be highly skilled in the art of packet analysis. Another concern is that SBID systems do not generate alerts that clearly state the issue at hand like a RBID system does. The data analyst will not receive alerts with a header "EXPLOIT dtspcd exploit attempt" for example. The alert will simply be a packet trace and it will be up to the operator to determine the nature of the alert. Furthermore, SBID system operators must establish an appropriate, effective threshold. Similar to determining the number of rules to implement in a RBID system, the threshold will either make or break the effectiveness of the SBID system. Set the threshold too high and the system will not alert on the necessary traffic, too low and the system will produce an overabundance of false positives.

For all the negatives though, a SBID system is still a valuable tool in the security toolkit. It is noteworthy that another form of anomaly detection called protocol anomaly detection helps to solve some of the issues plaguing SBID systems. Instead of training models on normal behavior, protocol anomaly detectors build models of TCP/IP protocols using their respective RFCs, although it is worth mentioning that protocol anomaly detection can also be built around traffic that is valid RFC traffic but is nevertheless anomalous traffic. (For more information, please see [RFC 791 \(IP\)](#) and [RFC 793, \(TCP\)](#) For more information on protocol anomaly detection read Kumar Das's "[Protocol Anomaly Detection for Network-based Intrusion Detection](#)".

## Layer, Layer, Layer

The optimal IDS configuration may be found in a combination of RBID and SBID systems. Rule-based intrusion detection systems are popular for a reason - they work. By keeping the signatures updated and utilizing a modified ruleset to suit the needs of the network, RBID administrators can detect 95% of applicable attacks. Unfortunately, it only takes one successful strike to cripple a network. Therefore, implementing a SBID system in *addition* to a RBID system may be a good solution.

By layering the intrusion detection systems, the network should benefit from both IDS approaches. The rule-based IDS will identify the known attacks and the statistical-based IDS will alert on most "0-day" and "slow and low" incidents. This configuration would have best detected the W32.SQLExp worm mentioned earlier. The behavior-based benefits of a SBID system would allow the security operator to identify the infected machine and take it off the network. The SBID system would recognize the unusual activity (excessive UDP packets on port 1434 and probes to random IP addresses) and alert on the anomalous traffic. Once identified, the worm would be easy for an operator to eradicate from the infected machines. Make no mistake, the best solution to effectively combating this worm would be the regular and diligent patching of systems. When new patches are not applied immediately though (and inevitably this time will come), a SBID system can provide a first response.

## Conclusion

There is still no IDS silver bullet. The best solution seems to be a combination of IDS approaches. There are a few vendors that offer the SBID system solution today. Fortunately, these solutions are all part of a larger offering that includes a RBID system. Rest assured, there will be bigger and badder worms than W32.SQLExp - in today's world of cyber-crime, malicious users, and cyber-terrorism, threats will undoubtedly continue to evolve and test security professionals. With the implementation of a statical-based intrusion detection system in addition to a rule-based system, though, you will be better protected against current and future threats. And maybe with the enhanced security on your network, you will be able to spend more time with your dog and less with a worm.

[Privacy Statement](#)

Copyright 2006, SecurityFocus