Intrusion Detection System

"Transforming Network Data into Security Knowledge."

White Paper

Rising to the challenge: Moving from Intrusion Detection to Security Monitoring

Author: Arrigo Triulzi
Chief Security Officer, K2 Defender Ltd.
Published: 8th July 2002

Introduction

Information security has been changing dramatically since the high-profile attacks on Yahoo! and eBay in 2000. Companies now rely extensively, perhaps excessively, on the Internet because it offers a relatively cheap and pervasive means to exchange information. Gone are the expensive point-to-point leased lines, in come the Virtual Private Networks running over the existing infrastructure of the Internet. With this saving in telecommunications costs comes a hidden cost: the exposure of a company's information infrastructure to the wild. The proliferation of ever more sophisticated firewalls, intrusion detection systems and Virtual Private Networks only attests to the mounting problems in trying to maintain the integrity of the Intranet.

Intrusion Detection has only recently made the headlines after high-profile sites and government agencies have made public their usage. It is also a relatively recent technology, back in the early 1990's the only real intrusion detection took place at the host level with the monitoring of access logs and application event logs or anti-virus programs on low-end desktops. The defence perimeter was often the host, perhaps slightly into the network with the introduction of ‘wrappers‘ around key network applications such as remote login support. Slowly, with the introduction of the World Wide Web, the interest in the actual traffic on the network increased, in particular as ‘site defacements‘ started taking place. Around 1995 the first real Network Intrusion Detection system was developed by the US military but it took until the last years of the century before non-military and non-academic sites started showing some interest. This change of mind was brought mainly due to the issues surrounding Y2K and, in particular, the thought that it would be the perfect smokescreen for an all-out ‘cyber-attack’. Fortunately it never materialised.

Developments in host-based and network-based intrusion detection started picking up once Y2K was over, sadly in response to an improved sophistication in the attacks themselves. The so-called ‘Black-Hat’ community has made quantum leaps in technology from the relative triviality of using debugging back doors in mail servers to the sophistication of packet crafting attacks subverting the normal structure of Internet protocols. Underlying these developments on both sides is one great weakness: the lack of an overall view of the security posture of an organisation. An enterprise might have the most sophisticated host-based intrusion detection tools, well-designed and configured firewalls and have deployed a network intrusion detection system but almost invariably is unable to see the overall picture. As if this disconnected picture wasn't enough, the constant growth in network size and capacity makes stand-alone network intrusion-detection solutions woefully inadequate.

Finally, large enterprises often have more than one site, perhaps on a global scale. It is not cost-effective to have separate monitoring centres and in fact it might be counter-productive as information sharing is often lacking. Being able to combine security information into a few monitoring centres provides a true instantaneous, enterprise-wide, view of the security posture.

This is where **K2-Defender** is positioned: developed from the ground up to be at the very core of a security strategy, combining high-speed, distributed, network intrusion detection with host-based intrusion detection into a single, network-aware, command and control centre.

Intrusion Detection

The concept of ‘Intrusion Detection‘ as applied to information systems was born in the days of mainframe computers and the need to monitor unauthorised accesses to those expensive systems. Within the context of high-end MIS installations intrusion detection was inextricably linked with what was, in effect, access management. Nowadays, Intrusion Detection has come to encompass every kind of security monitoring, be it network traffic or application access, at times even extending to anti-virus software.

The security community now subdivides Intrusion Detection Systems, often abbreviated IDS, into two distinct categories, Host-based and Network-based, abbreviated as HIDS and NIDS respectively. This subdivision is relatively recent and IDS on its own is still often used to mean NIDS. HIDS is used to describe all host-based (i.e. not necessarily network-aware) intrusion detection systems from the simplest log monitoring application to more sophisticated systems such as Tripwire™, for file modification monitoring and Portsentry™ for host connection monitoring. At times anti-virus products are also included in the HIDS category. NIDS describes all network-monitoring systems, whether active (such as firewalls with logging capabilities) or passive (such as network taps with packet processing facilities).

The key advantages of a HIDS is that it can have an intimate knowledge of the application being accessed and provide detailed information regarding the system being monitored. A NIDS instead has very little, if any, knowledge of the individual hosts but sees all the network traffic including attempts to reach applications on hosts which might not be monitored by a HIDS. For example, a NIDS can detect illegal network traffic but will not be able, except in particular cases, to detect repeated login failures on a particular application. A HIDS instead would detect the latter but be totally oblivious to attempts at circumventing network aware applications which are not under monitoring. This observation leads to the conclusion that it is generally best to have both HIDS and NIDS capabilities if a complete overview of the security posture is desired. An example of an IDS architecture, comprising both HIDS and NIDS with individual consoles is shown in *Figure 1*.

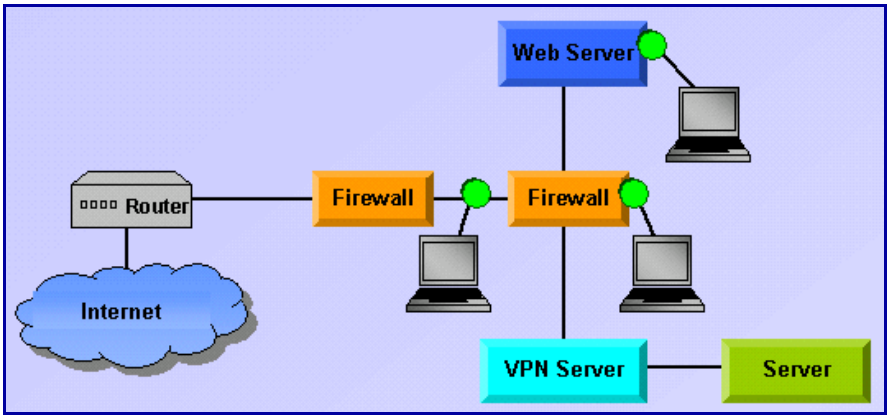

Figure 1: Traditional IDS Architecture

Unfortunately, like all monitoring systems, IDS technology can suffer from information overload. A single analyst cannot be expected to monitor multiple consoles for alerts and, should important alerts be flooded by false alarms, he cannot be expected to be able to react in a timely fashion. The problem of 'false positives' has always plagued the IDS community. It is impossible to avoid all false positives so the correct strategy is to work towards systems which minimise these occurrences.

False positives originate mainly from the IDS's inability to mediate an alert with respect to the environment in which it was generated. A lack of knowledge of both the network design and the hosts within it, means that often it will generate an alert based on the data examined out of the context in which it was produced. A simple example is that of an alert describing an attack designed to be effective against a Microsoft IIS web server that is being reported when the attack was targeting a Netscape Enterprise Server web host. In this particular example the attack, effective against a specific platform, is irrelevant against the platform being monitored. Without specific knowledge of the targets, better described as the 'context', it is impossible for an IDS to mediate the alert and assign it the correct threat level.

Once the false positive problem is addressed by providing the IDS system with knowledge of the context in which it is operating, the next issue is that of information overload. If a site has lots of traffic, both host and network based, the number of alerts, even genuine, will be substantial. Information consolidation is essential to avoid issues in which a large number of trivial alerts hides a single much more dangerous attack. The practice of creating a smoke shield around sophisticated attacks is widespread and needs to be guarded against. For an IDS system to be effective in these situations where the context is insufficient, there needs to be an aggregation mechanism which prevents large quantities of less-dangerous alerts from drowning highly suspicious attacks in the 'noise'.

An Overview of K2-Defender

The development of an Intrusion Detection system capable of addressing the concerns which we outlined in the previous sections has to start with the understanding that it needs to be 'enterprise capable'. This means that throughout the design the key criteria must be those of scalability, performance, availability, extensibility and manageability. Furthermore for a system of this size to be useful it must centralise the collection and presentation of data. This allows an enterprise to move from simple Intrusion Detection to *Security Monitoring* and policy enforcement. An example of how the network in *Figure 1* would be monitored is seen in *Figure*

2. Note in particular the large number of sensors giving a comprehensive view of the security situation.

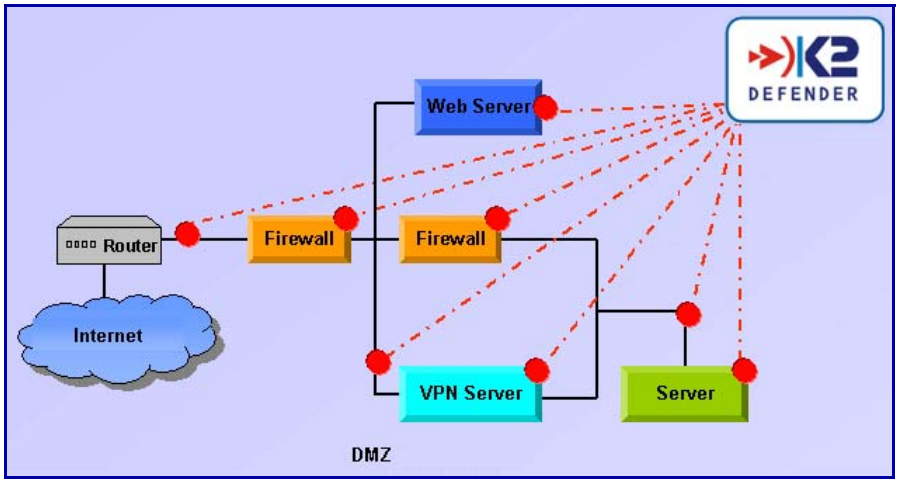

Figure 2: K2-Defender Network Monitoring

K2-Defender is a hybrid HIDS/NIDS system to enable correlation between what can be seen as 'local' events (HIDS) and 'global' events (NIDS) at a central console through a powerful database. Furthermore it is not limited by design to the monitoring of computer hosts but can also be extended to include logs from PBX systems or other computer-readable sources. The criteria outlined in the previous paragraph were the key driving forces behind the design of the system.

When considering scalability criteria it must be appreciated that enterprises grow and change continuously. Branch offices are moved, new offices are opened and networks change to reflect this. An enterprise class Intrusion Detection system must be able to respond to these changes with the smallest amount of effort. **K2-Defender** addresses this issue in two distinct ways: at the sensor level the installation and integration of new sensors is made as straightforward as possible, at the database level the tight integration with HP Himalaya *NonStop* technology leverages on the scalability of the database host.

Grow the System with the Security Perimeter

Scalability at the sensor level is achieved by allowing a network sensor to be added by simply connecting it to both the monitored network and the control network and turning on the system with a K2 sensor CD. The sensor will look for the database on the control network, authenticate with it, download both the code and rule set and enter operational status. Host-based sensors follow a similar procedure with the slight difference that a program is started as opposed to a system being turned on. At the host level the use of NonStop CORBA allows **K2-Defender** to scale with the host: simply adding processors, an operation which does not require any downtime on an HP Himalaya system, will increase the power of the system.

The issue of performance is crucial to avoiding 'black spots'. Just like high-security sites have closed-circuit cameras being recorded on time-stepped video recorders to avoid tape change blackouts, an Intrusion Detection system cannot afford to miss events. Performance is achieved by making use of multi-threading on sensors, to allow them to scale in performance on multi-processor systems and by optimising all data paths to ensure that the only constraint to monitoring is the pickup-point on the network or host. This means that to improve performance all that is needed is a sensor upgrade, perhaps from a Fast Ethernet to a Gigabit Ethernet adapter, or the addition of an extra processor to the sensor system. In addition, sensors can be configured in parallel on the same network segment, each applying a different set of rules. On the database side, performance is achieved by making use of all the advanced parallel database features offered by the HP Himalaya system. This guarantees that while data is being inserted into the database at high speed it can also be read and queried by analysts in parallel, a unique feature of the Himalaya architecture.

Security Threats don't Sleep

An Intrusion Detection system should be the last system to shutdown in an emergency like an extended power cut and the first to return to life after the incident. Monitoring distributed sites

places an even heavier burden as incidents local to the centralised monitoring site must not affect data collection at remote sites. With the current 'round-the-clock' model of operations in enterprises this means 24x7x365 guaranteed availability or '*5 nines*' availability with annual downtime measured in minutes. **K2-Defender**'s choice of the HP Himalaya platform, the only certified '*5 nines*' platform available, is a clear statement in support of guaranteed round-the-clock security monitoring. Throughout the system components are designed with the philosophy of the Himalaya architecture in mind: sensors never discard data until it is guaranteed to be committed to disk by the database, should communications from distant sensors be interrupted, data is held in buffers or on disk (depending on size) until communication can be re-established. No acknowledgement of data is sent on the host until it is committed to disk and guaranteed 'safe'.

Furthermore, the fast-restart capabilities of the Himalaya architecture mean that, should all power fail beyond the duration of on-board batteries and UPS systems, the system restarts from the exact point at which it lost power. This ensures that the core of the Intrusion Detection system is always the first to recover from such incidents.

Evolve the System with the Changing Threats

The 'Black-Hat' community is always developing new strategies. No Intrusion Detection system, no matter how powerful, can assume that it will be adequately monitoring a network six months after an installation. Extensibility is a crucial feature but in an enterprise this cannot come at the expense of availability. The design of **K2-Defender** includes support for both run-time reconfiguration and updating of sensors and the central database. Should IPv6 (the 'next generation' Internet protocol) become widespread, sensors can be upgraded by downloading suitable support. The upgrade requires no downtime: sensors are designed so that new code is started up in parallel to the old code and only when data is passing correctly through the new system is the old one shut down. This is not limited to sensors, support on the database host for the addition of new search mechanisms or data harvesting systems is also a run-time upgrade with no downtime. Extensibility is not only about new features but also about growing with the monitored domain. Sensors can be added and grouped at run-time, the central database simply making a note of the additional monitoring equipment and taking its data once it has successfully authenticated it.

Learn the Lesson from Centralised Network Management

All networks suffer from a management problem as they grow in size. Dispersed sites only add to the problem by perhaps being managed locally, independently of what might be a corporate policy. The only recourse in network management is to deploy systems such as HP OpenView. Security monitoring has not benefited until now from a similar system. **K2-Defender** recognises this issue and addresses it by making the whole system, both sensor and database, configurable and manageable from a single point. No matter how remote a sensor, its configuration is stored in the central database, along with that of all other sensors, and can be changed from the management console. Furthermore the configuration of the whole system is under change control, this means that when a new configuration is prepared and deployed it can be rolled-back should errors be encountered. This allows the configuration of the whole system, sensors and database, to be updated in parallel with a single change avoiding the issues related to different sensors being on different configuration releases. There is effectively no 'grey area' in which the configuration of the system is unknown or a mixture between old and new.

Just like configuration of distributed systems can be difficult, the deployment of new rules which describe what the system should alert on needs to be tightly controlled and coordinated. Within **K2-Defender**, rules are just a part of the overall configuration and as such benefit from the same parallel updating and configuration change control mechanisms.

The human interface to an IDS is where all the interaction takes place. It should be terse but informative and provide both real-time data and analysis tools. **K2-Defender** follows this philosophy by presenting real-time alerts continuously on the analyst's screen and allowing them to delve into the events which generated the alerts through search interfaces both at the alert and event level. Furthermore a wealth of statistical data is collected which is presented in graphical form as sometimes it isn't the alert which raises the suspicion of a problem but the unfamiliar 'shape' of statistical data on the traffic. The final part of the interface is a drag&drop report editor which stores all reports into the central database for cross-referencing and updating to minimise the risk of duplicate work and transcription errors.

An example of real-life deployment of **K2-Defender** is shown in *Figure 3* where the example is of an ISP with two co-location centres wishing to monitor its network traffic exclusively. Even without HIDS the centralisation of security monitoring allows for a single security team to monitor two geographically distinct locations.

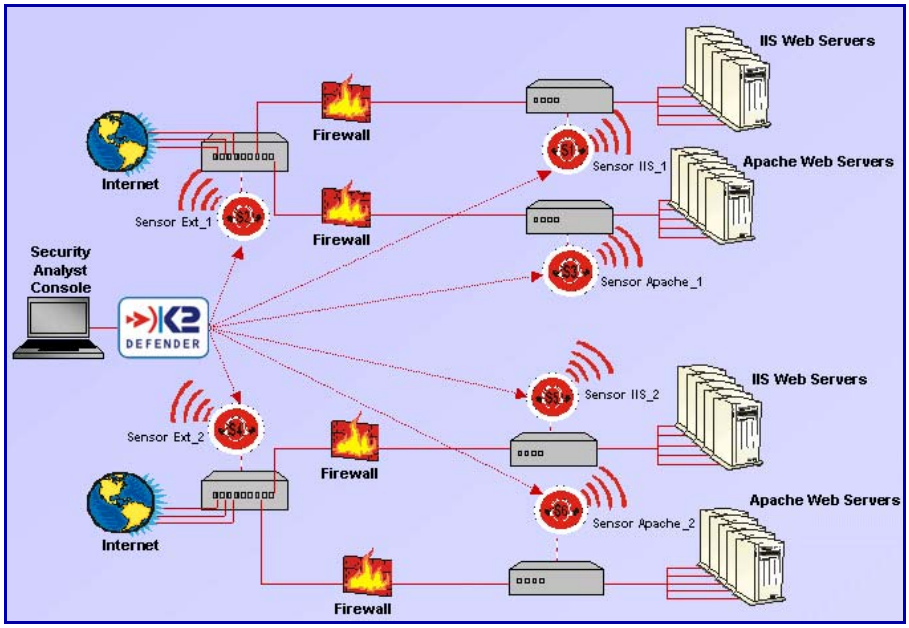

Figure 3: Monitoring of Twin Co-location Centres in an ISP

Harvesting Events

Sensor placement is one of the most hotly debated issues in Intrusion Detection. This is not because there is disagreement on where they should be placed in the first place but because the small number of sensors supported by a single console need to be placed judiciously to ensure maximum coverage.

K2-Defender leapfrogs the problem by supporting a number of sensors which is an order of magnitude larger than what was previously possible. The issue of whether to have network sensors before or after a firewall is moot: place one on both sides and perform *differential firewall analysis*.

Monitoring traffic at both sides of a firewall allows enterprises to confirm that the firewall is functioning correctly and that all the specified rules hold. This extends also to the conundrum of which internal networks to monitor. Instead of second-guessing the 'most dangerous' all can be monitored at the same time allowing the true traffic flows to define the danger levels. On a host by host scenario the situation is similar: once again why monitor a selection of hosts when all could feed the central database with data?

The advantage of a large number of sensors allows for a much more sophisticated monitoring technique. For example, beyond the differential firewall analysis, we can consider different rule sets being applied to the same traffic. This would be useful in the specific case where two different security groups are looking at different information and have differing requirements. A simple example could be a fraud prevention group looking at unauthorised document exchange while the security group would instead concern themselves with standard network attacks. Totally separate sets of sensors, feeding into the same database would allow both sides to correlate data but focus on their requirements in total security (thanks to the authorisation mechanisms for database access).

The combination of host- and network-based adapters provides the security analysts with unique correlated insight into the events of an enterprise network. The simple correlation of network traffic patterns with web log files allows analysts to determine whether an attack, viewed as a network traffic pattern from the NIDS part, has been successful or not as reported by the HIDS part on the web hosts. Sensor placement is no longer limited by the analyst's monitoring abilities but by consumables such as sensor hosts and network connectivity.

Turning Events into Security Information

Harvesting security events is definitely important but these events are of no use if there is no technology to help with the interpretation, they remain ‘events’ rather than ‘information’ useful for analysis.

Events in **K2-Defender** are clearly generated by sensors, both network- and host- based, but not only, internal processes within the central host also contribute. In particular ‘surveyors’, processes which trawl the database at low priority over a long time period searching for pre-defined patterns or peculiarities in the data.

With a system covering an enterprise, events can no longer be accepted individually and presented to the analyst; the information overload would overwhelm the analyst in the space of minutes. It becomes therefore essential to mediate all events and perform data-reduction exercises on them. Within **K2-Defender** no event is ever reported as a stand-alone event immediately. All events are mediated by an alerter which converts multiple events into single alerts. The power of this simple data reduction is immediately apparent in the trivial case of a heavy port scanning attack against the whole enterprise, perhaps comprising thousands of individual target hosts. Instead of receiving an alert for each and every port scan event against each individual target, a single alert is raised, containing all of the events and only this alert is displayed on an analyst's console.

For a more sophisticated example of a correlation between a network-based alert and a host-based alert. Imagine an attempt at gaining administrative privileges on a web server by means of a buffer overflow in a server-side script. There are two possible scenarios: the first is that the attack is not successful which will result in an error returned from the server, a ‘404’ error in web server jargon; the second results in an unfortunately successful attempt in which case some sort of success message is returned by the server, a ‘200’ message in server jargon. As far as a network-based sensor is concerned the event, if covered by a suitable rule, is the incoming buffer overflow attack towards a web server. From the host-based sensor the event of relevance is the addition of a line to the web log, either an error message or a success message. The two sensors taken distinctly provide only a fragmented view of the situation but both have a common part: the target. Both the network sensor and the host sensor will send a message which contains the same target address, the one of the web host under attack. At which point the correlation renders two independent events a single, interesting and informative alert: either an ‘attack unsuccessful’ or a more worrying ‘attack successful’ alert. **K2-Defender** ‘knows’ that the attack has been successful because it combines the network information describing a known attack with the lack of error message on the part of the server under attack. The same logic is applied to the generation of the failed attack message.

Have a Continuous Security Overview

The above example makes a powerful case for what is called in IDS jargon ‘*log fusion*’, the ability to merge different data sources into a single alert. An even more powerful case is made by the analysis of false positives. Let us take a different example, that of a load-balancing server from a remote web site continuously attempting to load-balance traffic to an enterprise network. Load-balancing servers are designed to attempt to discover the shortest path to a given client of a web site, often high traffic ones like news sites, by relatively intrusive means. By intrusive we mean methods which often result in alerts being generated by IDS systems. If we consider a large enterprise network there will be large numbers of these load-balancing attempts as employees all over the company access their preferred news source. Now, what if there was an attack which disguises itself as a load-balancing attempt? It suddenly becomes of paramount importance to be able to distinguish between legitimate load-balancing attempts and unsolicited ones.

A system which, like **K2-Defender**, offers log fusion would correlate data from the enterprise's web proxy servers of firewalls regarding outgoing web requests and data regarding incoming load-balancing events from the network sensors. A load-balancing attempt from a web site closely following a web page request from the enterprise network to the web site would trivially be discarded as a false positive and logged as an event to the database without alerting the security analysts. It is almost impossible to distinguish malicious use from non-malicious use of load balancing without resorting to log fusion.

Assess the Situation Before Crying wolf

Clearly there needs to be a distinction between the gravity of alerts. Not all alerts are of the same magnitude, especially in a large enterprise network with a multitude of different hosts. This is best exemplified by the simple situation of a Microsoft IIS web server and a Netscape Enterprise web server on the same network. The gravity of a Microsoft-specific attack against a Netscape Enterprise web server is negligible, barely above informational, whereas on the contrary the same attack against an IIS web server should not be ignored. This is defined within **K2-Defender** via a mediation mechanism which encompasses a knowledge of the network, obtained both by passive means and by configuration on the part of the security analysts.

Once alerts are received analysts need to be placed in a position to react to them. For this particular task **K2-Defender** has a dedicated object, a '*reactor*', which develops a response strategy once passed an alert. The strategy can be a combination of events such as an in-depth query of the historical database for a better definition of the risk or the recommendation to close a particular port on a firewall. Although there is a trend towards automated reactions, the danger of a crafted packet convincing the IDS to close all ports at the firewall and cutting off the enterprise from the Internet is too great. Any action which may result in a change of the network configuration is submitted to an analyst for approval or referral to the relevant network management team.

This describes the basic path which turns an event into an alert and eventually into a reaction, but often alerts are interesting not just on their own but viewed in context. This means that not only should single alerts be generated in response to multiple events but they should also be made context-aware before being delivered to an analyst. The simplest and yet particularly useful functionality which an analyst in a large enterprise desires is to be able to group alerts into classes, for example 'all alerts relating to web events'. This functionality is supported by **K2-Defender** beyond the simple grouping by alert type. Alerts can be grouped according to their origin, the alert type and the recipient. This allows alerts to be routed to the most competent analyst and to describe an escalation path should the alert not be handled within given timeouts. To support this, the analyst interface has explicit support for multiple 'alert channels', that is to say separation of the incoming real-time alert stream into a fully configurable set of channels.

Who? When? and most important, Why?

Often analysts need more than just alerts to comprehend the security situation. In particular previous knowledge of what has happened is of great interest. The equivalent of analyst experience in IDS terms is a historical record of all events. This has been available, in different ways, since the very first IDS systems but searching through it has always been laborious to say the least. **K2-Defender** improves upon previous designs by offering an interface to the database which stores all events and by offering 'surveyors'. These are background processes running at batch priority which continuously trawl the database looking for patterns, small but possibly significant events and user-requested analysis. The alerts generated by surveyors are processed like all others but are clearly not real-time. They are designed to answer questions like 'have I ever seen a sequence of alerts like this one before?' which is more than just a simple query for packets or log entries as it is in reality a query for a list of alerts or specific events.

Surveyors lead into forensic analysis. It isn't always possible to prevent attacks, the bane of all analysts being so-called '*0-day exploits*'. These are attacks which have been discovered but not yet published in an open or semi-open forum making it impossible to know what exactly they involve. They are in effect surprise attacks and they are almost impossible to guard against. Sadly the only recourse after being a victim of a 0-day exploit is to perform forensic analysis to understand the vulnerability and guard against it in the future. To perform forensic analysis, the larger the amount of data the more accurate the analysis. This makes **K2-Defender** an ideal platform for the task, because in a single location all events and alerts are available for analysis. Sometimes it isn't the host-specific information which explains a 0-day analysis but the overall context of the network. An example might be a firewall bypass incident due to a mis-configured router which allows user-specified routing into the network. Packets with user-specified routing have a distinctive signature which a HIDS would not be able to pick up. But a NIDS sensor on a remote network through which the attack entered might have recorded the packet and, in conjunction with the HIDS logs recording the successful intrusion, explain the origin of the vulnerability.

The most labour intensive and error-prone task for an analyst is the preparation of an incident report. This needs to contain all the relevant data, analysis and ancillary information for a Chief

Security Officer or Chief Information Officer to take an informed decision. Furthermore it is desirable for the reports to be available for future reference: a *Microsoft Word* file sitting on a remote analyst's laptop is of little use to the enterprise-wide security effort. The easiest solution is to ensure that all reports are centralised onto the database and that data is not simply copied but referenced so that multiple reports can be linked to the same data, and through that to each other. This is precisely what **K2-Defender** implements: all reports are held on the central database in an industry-standard format (XML), cross-referenced on the basis of the data linked to them. Report editing is provided via a report editor which supports drag&drop of data onto the report.

Security Provisions within K2-Defender

All security systems are prime targets for attack by sophisticated intruders. This is clearly because the blinding of countermeasures makes the rest of the intrusion so much simpler. With this in mind **K2-Defender** was designed with stringent security requirements. As with all security designs the aim is to 'raise the bar' sufficiently high that only the most dedicated of intruders will attempt a break-in.

Throughout the system all remote communication is authenticated and authorised. Authentication takes place both at startup between sensors and the central database and during operations by means of encrypted heart-beats. Should any sensor cease to reply to heart-beat requests or provide an incorrect response to the encrypted challenge the central database would disable said sensor and notify the administrator of the event. Object authorisation is handled by the central database based upon unique keys shipped with each sensor on read-only media which need to match with those shipped with the system. Unauthorised sensors are not allowed to connect to the central database.

The constant stream of heart-beat requests is also designed to foil basic traffic analysis of sensor-database communications. Sophisticated traffic analysis will be able to collect enough useful data to understand data trends but will require substantially more effort than what might be normally available.

All timestamps are kept from signed, trusted Network Time Protocol clocks. External sources are not trusted, one or more GPS-locked network clocks are provided using RSA public key encryption to exchange keys with the various **K2-Defender** objects.

No objects can communicate with networks outside the K2 control network except through gateways which authenticate and authorise the transactions on a per-method basis. Users which are not authenticated have no access to gateways as users who have insufficient authorisation for a specific method. Data on the system is also subject to authorisation and authentication both with respect to analysts and actual other objects in the system. Should data be restricted to a particular group of objects or analysts, access will be blocked from other groups.

Finally, all data is both shipped from the sensors and stored on the database with a cryptographically strong hash generated using the public key of an external trusted third party to prevent claims of data fabrication or modification.

Conclusion

The evolution of 'Black-Hat's' tactics leaves very little scope for evolving stand-alone solutions. One of the biggest weaknesses in any organisation is the lack of communication which becomes fatal during security events. At times of emergency it is important to be able to find everything which is needed to take educated decisions in the face of the threat being experienced. Stand-alone solutions which might be excellent in their local domain fail when seen within the larger picture of an enterprise's security posture.

K2-Defender is more than a HIDS/NIDS hybrid system, it is the core component for the distributed monitoring of sophisticated IT infrastructures. The design itself, from the ground up, is built around the core concepts of scalability, availability and consolidation of information to overcome the limitations of local solutions. Support for prosecution makes it into a powerful tool not only for the deterrence of internal fraud, but also for the assistance of law enforcement agencies requesting security records.

K2-Defender is the first step into the complete integration of all the elements of an enterprise security strategy.

Intrusion Detection System

For further information contact:
K2 Defender Ltd.
Garden House
Cloisters Business Centre
8, Battersea Park Road
London SW8 4BG, UK
http://www.k2defender.com

P/N: K2DWHPA003

Part Number: K2-Def-WHPA-IDSM-001