

Solution Brief

Using Compound Signatures to Protect Against Complex Attacks

*Accurately Identifying Attacks with Juniper Networks
NetScreen-IDP Product line*

Sarah Sorensen
Product Marketing Manager



Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
408 745 2000 or 888 JUNIPER
www.juniper.net

Part Number: 351039-001

Contents

Introduction.....	3
What is a Compound Signature?	3
Using Compound Signatures to Efficiently Identify Attacks	5
Using Compound Signatures to Stop Peer-to-Peer Network Activity	7
Ordered versus Unordered Events.....	8
Conclusion	12

Introduction

Juniper extends its innovative attack identification capabilities with the introduction of Compound Signatures. This new detection method is optimized to detect very complex attacks in an organization's network and is part of the latest release of Juniper Networks NetScreen-Intrusion Detection and Prevention (IDP) system, version 3.0. With Compound Signatures, Juniper combines nine different mechanisms in Multi-Method Detection to provide customers the most comprehensive attack protection capabilities on the market.

Compound Signatures augment existing detection methods by giving an administrator the ability to chain together events and look for multiple anomalies and attack patterns, either sequentially or in parallel. For customers, this means Juniper Networks NetScreen-IDP delivers a new class of signatures that are ideal for capturing a cause and effect relationship between two entities on the network and taking into consideration exponential risk associated with multiple incidents to ensure accurate, efficient detection of complex attacks.

This security "Solution Note" will describe what Compound Signatures are and how they can be used to protect against complex attacks.

What is a Compound Signature?

Compound Signatures leverage components of Juniper Networks NetScreen-IDP's Protocol Anomaly Detection and Stateful Signatures to further refine the attack detection search process to pinpoint complex threats.

Stateful Signatures are able to detect known attacks, looking for attack pattern (signature) matches in the relevant service field contexts of the protocol stream. Protocol Anomaly Detection is a complementary technique able to identify some unknown attacks and exploits that cannot be characterized with an attack pattern string, using prior knowledge of network protocols to search for deviations from normal use as defined in protocol specifications.

While both methods provide vital attack protection capabilities, they can, at times, be limited in their ability to efficiently detect some complex attacks. Stateful Signatures are uni-directional by definition, matching a single pattern on a given context. For attacks best identified by matching a series of patterns or a cause and effect relationship, Stateful Signatures are too reductionistic. While Protocol Anomalies are able to check client and server replies and hence provide some means of modeling the network traffic to identify some multi-part attacks, they can be too generic in nature for some complex attacks.

Compound Signatures are designed to combine multiple Stateful Signatures and/or Protocol Anomalies to optimize the search criteria to incorporate more than one attack characteristic to ensure accurate and effective identification. They are built on the concept of modeling sequences of actions between two actors (client and server), which is a well-

tested and common paradigm in computer science.

For example, to implement an algorithm in a programming language, the programmer would code something along the lines of:

```
If (some event occurs) and (some other event
occurs) and ...
    Then (take the appropriate action)
Else
    Do nothing
```

Essentially, Compound Signatures give customers similar functionality, but in Juniper Networks Netscreen-IDP the sequence of events can be defined to occur both in parallel, as well as in ordered sequences (as in the above example).

The benefits that arise from this new functionality are profound:

- Making it possible for the system to identify a cause and effect relationship that signifies malicious intent
- Increasing accuracy of complex attack detection
 - Instead of being tied to a single attack string, the system can weigh the combined consequences of matching multiple anomalies and/or patterns in the attack identification process
- Allowing the creation of more specific signatures for application protocols not natively supported in IDP
- Enhancing processing performance
 - In comparison to stream signatures, which previously had to be used to detect similar activity

Using Compound Signatures to Efficiently Identify Attacks

What Compound Signatures bring to the table is an abstraction that increases the robustness of IDP's detection capabilities. By modeling network penetrations as sequences of events that must occur before an attack is flagged, IDP is easily geared towards "cause and effect" situations. This can be ideal for parsing full-duplex traffic between two network entities (e.g. client-server and P2P networking), which is important, since network traffic that results in a compromise in network security often arises from security holes prevalent amongst client/server protocols.

In the specific examples that follow, signatures could be used to detect and, thus, prevent the attacks, however, Compound Signatures represent a more efficient and accurate way to identify these multi-part attacks. The reconstruction steps that the Juniper Networks NetScreen-IDP system utilizes to reduce ambiguities in the application message is one means to reduce irrelevant pattern matches, and the ability of its Stateful Signatures to pinpoint an attack pattern match to only the relevant service field is another. Now, with Compound Signatures the system can go one step further and chain together potentially suspicious activity to detect the malicious traffic with certainty. Plus, for those protocols not natively supported in Juniper Networks NetScreen-IDP, administrators can now efficiently look for attacks. Rather than matching a string to a stream, IDP can break down attack matches into smaller sequential chunks that can accurately pinpoint and detect an attack.

For example, consider the case of protecting against an SMTP vulnerability that affects only a certain version of an SMTP service. It is more advantageous to identify an attack exploiting this vulnerability by looking for:

```
If (you see SMTP network traffic) and
    (this traffic pertains to the xxx version of the SMTP
service) and
    (you search for and find this vulnerability signature)

    Then (take appropriate action)

Else
    Continue
```

than to perform the equivalent functionality using stream signatures that would look for a superset attack pattern within the entire client/server communication stream. In the above example, a client could initiate the traffic, the server would then need to respond with traffic containing a string denoting the particular SMTP service and version number, and sometime later, the client transmits traffic containing the malicious string. Due to the amount of network traffic attributed to this sequence of events, it is not practical to define a single Stateful Signature to capture this behavior; but with Compound Signatures, it now becomes simple to command the IDP system to look for precisely this traffic within this sequence of events.

Taking the e-mail vulnerability example one step further, consider protecting your network against the W32/Mimail attack (http://www.cert.org/incident_notes/IN-2003-02.html). This attack is propagated by an e-mail with a seemingly innocuous and common title, along with a specially crafted zip file named message.zip. When extracted, the file surreptitiously installs malicious code on the host, which will start mass mailing from that machine. Utilizing a stream signature to trigger on the e-mail title is not a viable strategy, since the possibility of false positives is too large. Triggering on e-mail attachments named message.zip is also a sub par scheme, since it too has a high probability of generating numerous false alarms that would render that protection useless.

Attackers are becoming more sophisticated in their methods, using the limitations of today's attack detection capabilities to their advantage and employing tactics, such as those used in the Mmail attack, to evade legacy solutions. With Juniper Networks Compound Signatures, however, IDP can use a signature that states if an e-mail is encountered with some particular e-mail title *and* if said e-mail contains an attachment name message.zip, then the probability that the e-mail is malicious in nature is high and the e-mail can be blocked. In essence, Compound Signatures gives the user the ability to more concisely describe attack scenarios, which provides IDP with more information to match against when filtering voluminous amounts of network traffic to ensure attacks are accurately and efficiently identified.

Using Compound Signatures to Stop Peer-to-Peer Network Activity

While Juniper Networks NetScreen-IDP provides broad Peer-to-Peer (P2P) protection, there are always new specific implementations introduced or evolved that need to be incorporated into an organization's security stance. As a result, Juniper Networks Compound Signatures offer an effective tool for protecting against these specific implementations, even if the protocols are not natively supported in IDP.

The dynamic nature of the protocols makes preventing or controlling P2P network activity a challenge. P2P application protocols are quite adept at hiding their tracks, operating directly on top of TCP or UDP with the ability to work over any port.

The Direct Connect file sharing service is one such P2P application, which is capable of sending and receiving data over any open port. In order to prevent a client from logging into a Direct Connect file-sharing hub, Juniper Networks NetScreen-IDP must be on the lookout for the handshaking protocol to initiate the session. The way this works is that the client wishing to connect to the P2P network sends its "nickname" to the hub using a well-defined packet format (the string "MyNick" followed by the nickname). The hub responds with a packet containing the string "Lock," which is used by the client to generate a key, who then responds in turn. Searching for this pattern using a stream signature approach runs the risk of firing too many false alarms to be useful. As Stateful Signatures are unidirectional in nature, the only way to search for this P2P traffic is to look for packets with the string "My Nick" or packets with the string "Lock," which is clearly not a precise strategy.

However, with Juniper Networks Compound Signatures the problem becomes easier to describe, model, and implement. With Compound Signatures, the sequence to look for can be described as such:

1. Look for the string "Lock" emanating from any server
2. Client responds with an encrypted key denoted with a packet containing "Key"
3. Server responds back with packet containing the string "HubName"
4. Client responds back with packet containing the string "Version"

In this way the detection of the attack is split into smaller chunks of work, with added dependencies between these chunks. In effect, IDP is able to describe in a concise manner the communication pattern between two network entities. This Compound Signature will be highly accurate, and provides thorough protection against an application protocol not natively supported by Juniper Networks Netscreen-IDP.

While these P2P handshaking protocols are well known, they do not lend themselves to easy detection. Direct Connect is just one example. There are numerous others in active use and most of them are very good at covering their tracks. In order to hide themselves, P2P handshaking protocols use techniques that just cannot be monitored in a feasible manner using stream signatures. However, by using the functionality provided by Compound Signatures, in conjunction with the knowledge of how the underlying P2P protocol works, it is now practical to efficiently prevent such activity on the network.

Most of these handshaking protocols follow a similar pattern, with the client initiating a connection by transmitting a small (2 or 3 byte) message, followed by fixed-length data, oftentimes containing an encrypted key. The P2P file server or hub responds with another small message, followed by fixed-length data needed to maintain the session. It is simply not feasible to apply a 2 or 3 byte signature to each incoming packet because the potential for false positives is too great. As in the case of Direct Connect, compounding the problem is the fact that almost all of these protocols operate over any port, making a signature-based approach untenable in many instances. But since the packet lengths are fixed, even though much of the P2P traffic is encrypted (which obviously rules out signatures as a viable course of action), a clear chain of events begins to emerge. Such structured traffic patterns forms a dependency for which Compound Signatures are perfectly suited for describing, leading to a solution that achieves high accuracy, even for protocols not natively supported by IDP.

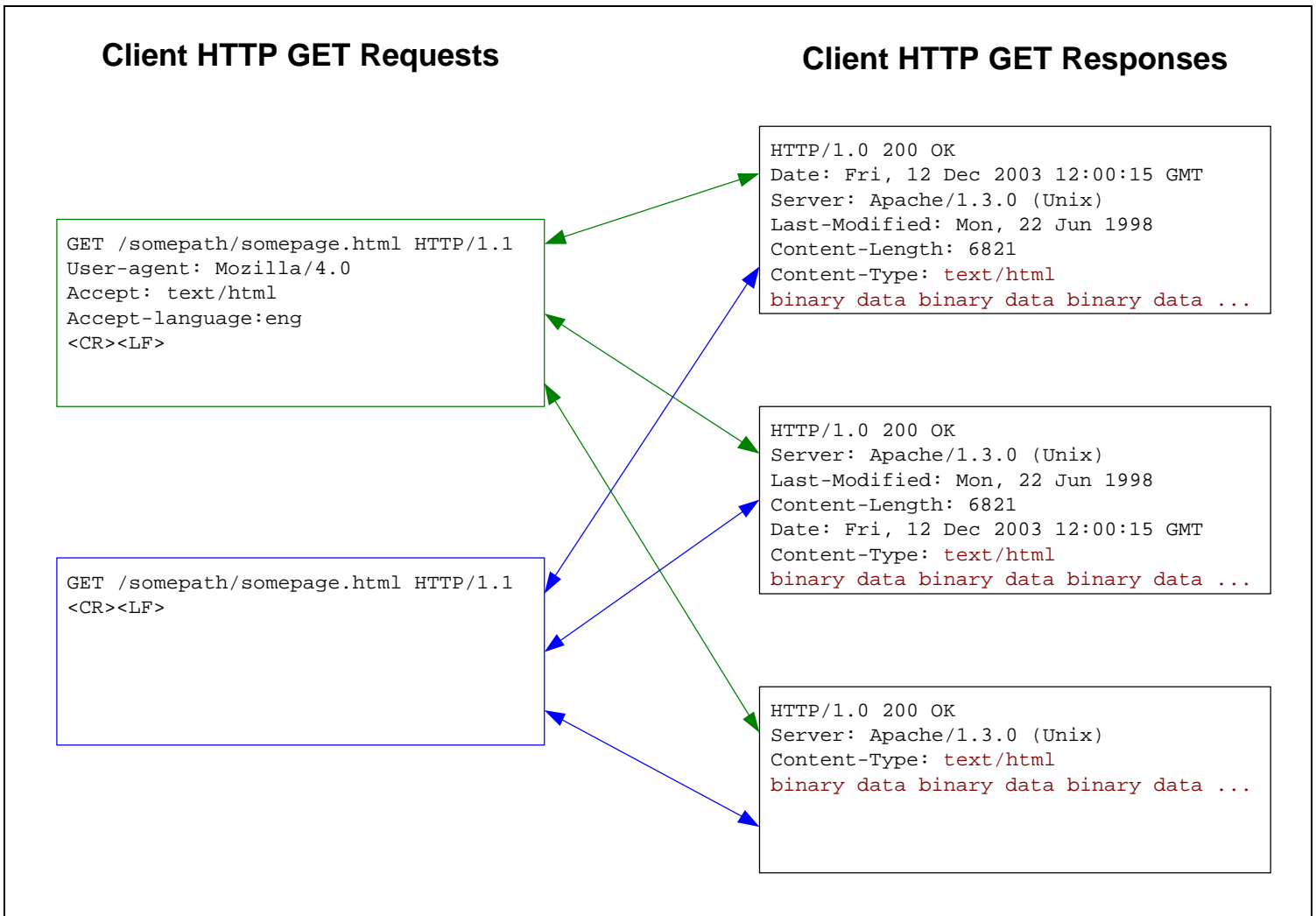
Ordered versus Unordered Events

Compound Signatures can have up to 32 individual members, and each member can be either a Protocol Anomaly or a Stateful Signature. All the constituent parts of a Compound Signature must match before an attack matches, but the chain of events may be ordered or unordered, which is quite important for certain scenarios.

Suppose you want to protect your network from the dubious scenario where the payload of an HTTP response to a GET request for an HTML page contains binary data. This situation is highly suspicious since HTML is a character-based markup language and there should not be any binary data in the response. If the response from the Web server indicates that the payload content is supposed to be HTML, but the HTTP data is binary in nature, then this characteristic makes the session suspect and malicious in nature. Such a payload may indicate a buffer overflow attack (where the binary data is malicious code) or some other subterfuge. In general, the format for HTTP requests consists of an initial line (GET, POST or HEAD) followed by a set of header lines. The HTTP protocol does not specify the order in which these header lines must appear and, moreover, in many cases they are optional and may not appear at all in the request or the response.

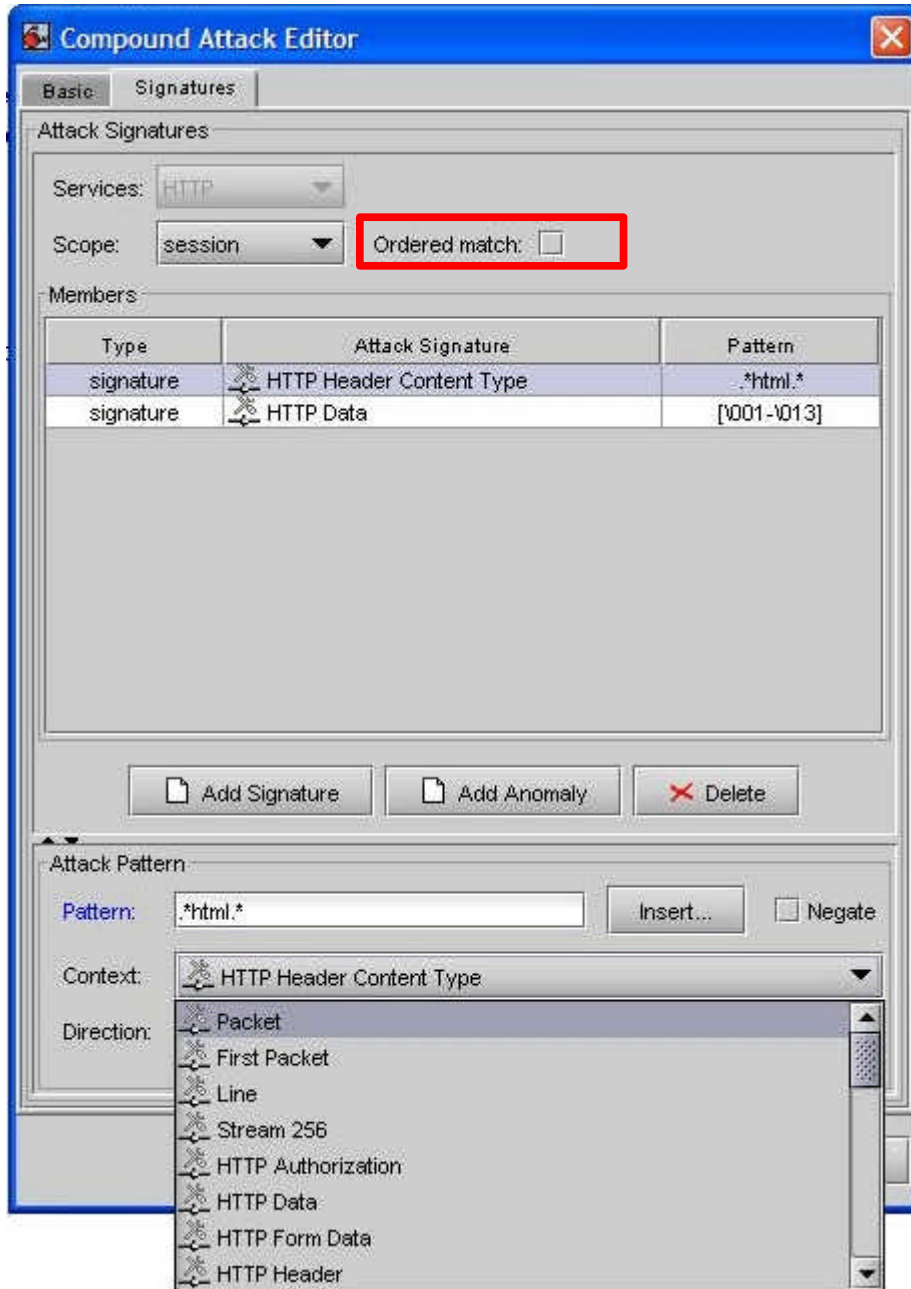
Figure 1 depicts a few possible scenarios for this example. Here, in order to deploy an effective attack protection, IDP needs to efficiently handle all six of these permutations, in addition to the many more that exist just for this contrived example. It is difficult to accomplish this using a signature-based approach, but by generating a composite signature using an unordered sequence, IDP can be configured to guard against this and countless other similar attacks.

Figure 1:



Compound Signatures are configured via the Compound Attack Editor in Juniper Networks NetScreen-IDP Manager. Figure 2 is a screen-shot from the Juniper Networks NetScreen-IDP Manager showing how the administrator can configure their device to protect against this HTTP attack. Here, Juniper Networks IDP is instructed to flag an attack if the content type in the server response contains the string "html" (i.e. the header in the HTTP packet looks something like "Content-Type: text/html") *and* following that, the payload contains binary data.

Figure 2:

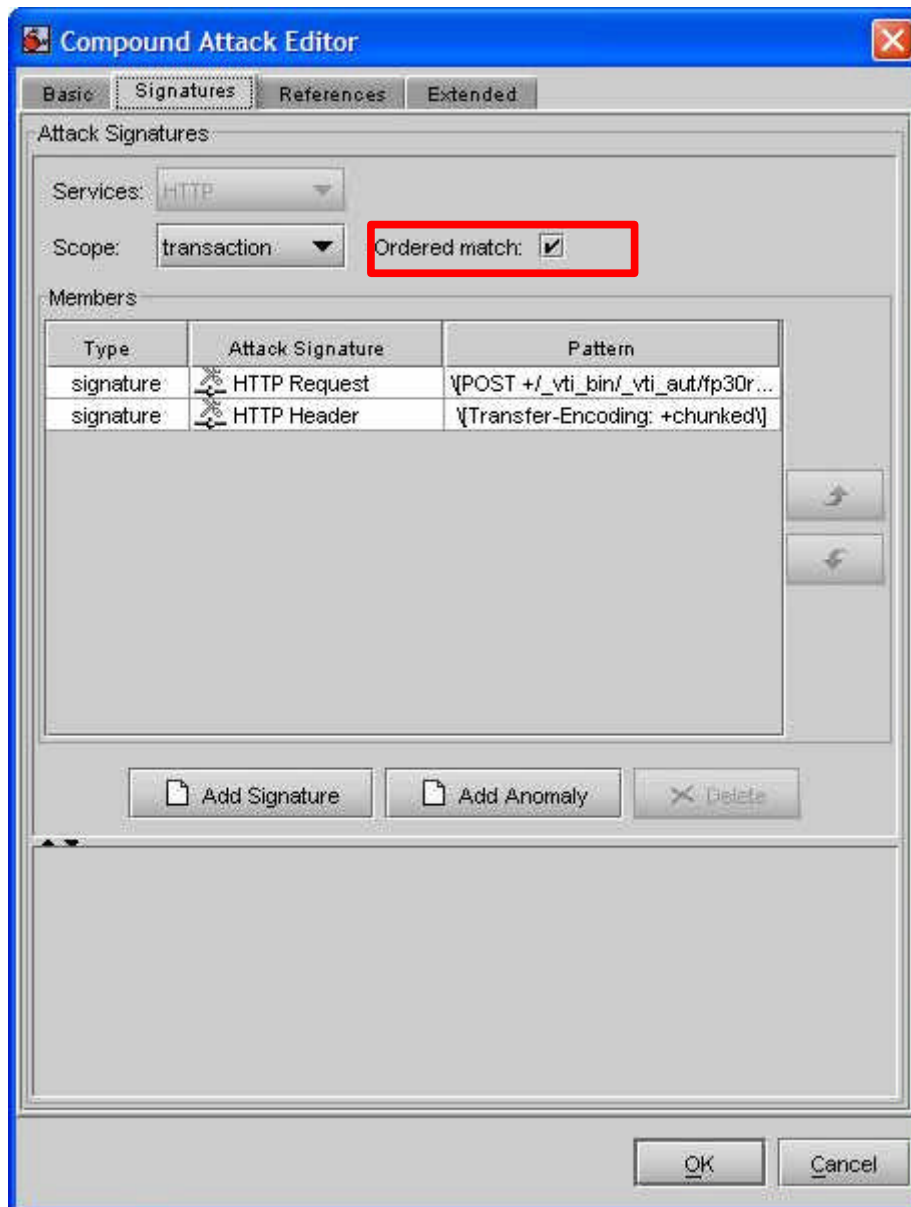


Note that there is a check box for specifying whether the events constituting the compound attack must be ordered or un-ordered.

If this checkbox is selected, then the Compound Signature is an ordered match, in which case arrows appear on the right that are used to lay out the events in the specific order in which they must appear before a sequence of events is flagged as an attack. Figure 3 is an example of an ordered Compound Signature, in this case the compound signature is protecting against a “Chunked Encoding Post” vulnerability, which is a DoS attack described in full detail at:

<http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS00-018.asp>:

Figure 3:



Without Compound Signatures it is very difficult to capture this type of complex and interdependent behavior between the HTTP client and the Web server. By adding additional context to the network traffic pattern, Compound Signatures allows Juniper Networks NetScreen-IDP to search for attacks in the most efficient manner possible.

Conclusion

Compound Signatures represent a new tool in the never-ending battle against increasingly sophisticated layer-7 attacks. The need to capture, model, and efficiently analyze dynamic traffic between two entities on the network is clear. With Compound Signatures, the Juniper Networks IDP processing engine now has at its disposal even more information about the context of the network traffic and can make more complex determinations of what represents truly malicious intent. With this new class of signatures, the Juniper Networks Netscreen-IDP product line continues to advance the state of the art in detection algorithms, bringing benefits to the end user in the form of increased robustness, system accuracy, and the highest level of protection against a range of complex application-layer attacks.

Copyright © 2004 Juniper Networks, Inc. All rights reserved.

Juniper Networks, the Juniper Networks logo, NetScreen, NetScreen Technologies, GigaScreen, and the NetScreen logo are registered trademarks of Juniper Networks, Inc. NetScreen-5GT, NetScreen-5XP, NetScreen-5XT, NetScreen-25, NetScreen-50, NetScreen-100, NetScreen-204, NetScreen-208, NetScreen-500, NetScreen-5200, NetScreen-5400, NetScreen-Global PRO, NetScreen-Global PRO Express, NetScreen-Remote Security Client, NetScreen-Remote VPN Client, NetScreen-IDP 10, NetScreen-IDP 100, NetScreen-IDP 500, GigaScreen ASIC, GigaScreen-II ASIC, and NetScreen ScreenOS are trademarks of Juniper Networks, Inc. All other trademarks and registered trademarks are the property of their respective companies.

Information in this document is subject to change without notice.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without receiving written permission from:



Juniper Networks, Inc.
1194 N. Mathilda Ave. Sunnyvale, CA 95014 ATTN: General Counsel