

SUN AND CLOUDMARK STOPPING MESSAGING ABUSE IN SERVICE PROVIDER ENVIRONMENTS

White Paper

April 2007

Abstract

Service providers today face increasingly sophisticated spam, phishing, and virus attacks that impact messaging infrastructure and pose serious security risks that result in lost revenue and subscribers. The combination of Cloudmark Authority software and the Sun Java™ System Messaging Server running on Sun platforms provides a high-speed, highly accurate messaging security solution. This paper describes the scalability, performance, and availability features built into the solution design that help support an enterprise-wide, carrier-grade messaging implementation for service providers and enterprises.

Table of Contents

Introduction	1
Scalability	1
Performance	2
Availability	2
The Cloudmark and Sun Solution	3
The Solution Architecture	3
Cloudmark Authority and Sun Java System Messaging Server Logical Architecture	4
The Cloudmark Message Filtering Mechanism	5
Cloudmark's Automated Abuse Tracking System	5
Cloudmark Micro-Updates Service	5
The Sun Java System Messaging Server and Cloudmark Network Architecture ...	6
Handle More Subscribers, Gain Accuracy and Performance	8
Greater Filtering Engine Scalability	8
Scaling the Feedback and Update Architecture	8
More Performance	9
Extremely Efficient Algorithms	9
Fast Hash-Based Cache Lookups	9
Optimal Algorithm Coding	9
Speedy Attack Response	10
Authority Performance Statistics	10
Highly Available	10
Stateless Architecture	10
Resistant to Failures	11
Resilient Client-Server Architecture	11
Load Balancing to Increase Availability	11
Handling Failure Conditions	12
Sun and Cloudmark	12
For More Information	13

Introduction

The global reach of the Internet heightens the threat of increasingly sophisticated spam, phishing, and virus attacks, posing serious security risks for fixed and wireless operators and resulting in lost revenue and subscribers. As messaging abuse grows, costs escalate, satisfaction plummets, and customer churn rises. Service providers must find ways to solve the messaging abuse problem while at the same time reducing costs. The Cloudmark messaging security software running on Sun platforms offers a powerful solution that minimizes the impact of messaging abuse on the business.

Service provider environments demand enterprise-wide, carrier-grade implementations capable of supporting business-critical applications and services. Solutions deployed in this type of environment must support 7 x 24 operation and scale to meet demands. The combination of Cloudmark messaging security software and the Sun Java™ System Messaging Server, running on Sun platforms, offers a powerful solution that minimizes the impact of messaging abuse on the business. The Cloudmark and Sun solution offers a unique approach to three critical capabilities vital to carrier-grade service provider environments — scalability, performance, and availability. This document discusses how the solution addresses these three needs to provide enterprise-wide, carrier-grade implementations suitable for service providers.

Scalability

Scalability is perhaps the most important feature when evaluating software solutions, particularly for applications such as messaging security. Scalability describes the ability of a system, network, or process to handle growing amounts of work in a graceful manner, or to be readily enlarged¹. There are several areas within messaging security software where scalability is a factor, including:

- *Filtering volume*

The filtering mechanism must be able to scale to handle heavy increases in messaging volume, either through increased legitimate traffic such as additional subscribers, or through an increasing number of threats. Spam, viruses, and phishing comprise a significant portion of the tens of millions of messages sent and received by messaging providers today. Many messaging security products are unable to handle high message volumes and growing numbers of users at service providers and other large enterprises.

- *System growth*

It is important to be able to increase the total system size in a clean and manageable way in response to the growth of the subscriber base. Software that cannot scale efficiently can increase the need for system administrators and generate additional workloads.

1. Andre B. Bondi, 'Characteristics of Scalability and Their Impact on Performance', Proceedings of the 2nd International Workshop on Software and Performance, Ottawa, Ontario, Canada, 2000, ISBN 1-58113-195-X, pages 195-203

cannot scale efficiently can increase the need for system administrators and generate additional workloads.

- *Adaptable response*

Messaging security software must include a scalable process that can respond to the constantly morphing waves of malware that afflict carrier systems. Many systems utilize manual intervention and cannot handle increasing numbers of carrier-class customers and the rapidly escalating volume of abuse.

Performance

In order for messaging service providers to host a service that can adapt as needs change while still containing costs, the service must provide suitable performance. In the context of software applications, performance describes the efficiency with which the application reacts or fulfills its intended purpose. There are several aspects of software performance that are important to messaging environments:

- *Raw performance*

If an application displays good raw performance, it can run efficiently on a cost-effective hardware infrastructure, thereby reducing initial capital expenditures, decreasing operational costs over the life of the solution, and providing an optimal return on investment (ROI).

- *Predictable performance*

Raw performance alone is not sufficient if it cannot be sustained in the face of rising messaging abuse threats. Predictable performance is absolutely essential to service providers, both for capacity planning and budgetary purposes.

- *Response performance*

The system must be able to respond quickly to new and evolving attacks, including those that are increasingly transient, such as phishing.

Availability

Availability is the third feature for service providers to consider in evaluating carrier-grade messaging solutions. A key metric for customer satisfaction and service level agreements (SLAs), availability describes the percentage of time the software is running correctly. Poor availability can have a direct impact on the bottom line, and is a critical requirement for carrier-class software. When considering software integrated as part of another service, such as the Sun Java System Messaging Server, there are many facets that comprise availability. Several important areas include:

- *Design*

The only way to ensure that application availability is reliable and consistent is to design it into the framework of the software, rather than tack it on as an afterthought.

- *Multiple failure scenarios*
The software must be flexible enough to handle many types of failures, including transient as well as hard failures.
- *Suitability*
The solution architecture must be tailored to the task at hand, integrating appropriately with any host software. For example, load balancing can be a better approach than clustering when handling stateless requests.

The Cloudmark and Sun Solution

Cloudmark software running on Sun systems provides a high-speed, highly accurate inbound and outbound spam, phishing, and virus detection solution for service providers. Unlike traditional email security applications, Cloudmark technology does not utilize rules or inaccurate blacklisting methods for email filtering. Instead, the Cloudmark software utilizes advanced algorithms to examine email and wireless messages, creating a unique set of identifiers, or *fingerprints*. Using this sophisticated fingerprinting technology, Cloudmark software addresses the evolving nature of messaging threats by adapting and catching new threats as they mutate.

The Solution Architecture

Based on best-of-breed products from Sun and Cloudmark, the solution consists of the following key components:

- *Sun Fire™ T1000 and Sun Fire T2000 servers with CoolThreads™ technology* — Sun Fire servers incorporate an optimized chip multithreading capability that provides an incredible boost in performance for Web servers, databases, and other applications that must move vast amounts of data on and off networks quickly. Capable of attaining throughput levels optimally suited to service provider environments, Cloudmark's service provider customers have experienced message throughput on Sun Fire T1000 and Sun Fire T2000 servers to be almost twice that of other hardware solutions.
- *Sun Java System Messaging Server* — The Sun Java System Messaging Server is an industry-leading, high-performance, highly secure messaging platform utilized by service providers. Capable of scaling from thousands to millions of users, the Sun Java System Messaging Server is ideal for consolidating email servers and reducing the total cost of ownership of communications infrastructures.
- *Cloudmark Authority software* — Cloudmark Authority is the leading messaging infrastructure security software for service providers. Using unique, advanced filtering technologies, the Cloudmark Authority software tracks threats in order to detect spam, phishing and virus mutations before they propagate.
- *Cloudmark Micro-Updates Service* — The Cloudmark Authority Engine contacts the Cloudmark Micro-Updates Service by default once every minute to retrieve new

fingerprints stored in Cloudmark's database of known abusive messaging fingerprints. The fingerprints are loaded into memory and used to evaluate new incoming messages. The Cloudmark Network Feedback System funnels reports from the Global Threat Network back to the Cloudmark System.

- *Cloudmark Global Threat Network* — Made up of 180 million trusted users who give feedback on incoming messages, the Cloudmark Global Threat Network sends fingerprints for a message to the Cloudmark Micro-Updates Server for distribution to Cloudmark Authority installations if enough users in the network report a message as abusive. This protects Cloudmark users within minutes of a new abuse outbreak.

Cloudmark Authority and Sun Java System Messaging Server Logical Architecture

Figure 1 shows the logical architecture of a Sun Java System Messaging Server installation incorporating Cloudmark Authority software for messaging security. Using the Cloudmark Anti-Abuse Client (CAAC) Plugin, the Sun Java System Messaging Server Message Transfer Agent (MTA) directs incoming messages to the Cloudmark Authority Engine using the TCP/IP protocol. The Cloudmark Authority Engine Server receives each message, evaluates it, and decides whether or not the message is abusive. The MTA is able to make use of attributes from the Sun LDAP directory to decide whether to invoke the Cloudmark Authority Engine for a particular user, or whether to invoke the service for all users.

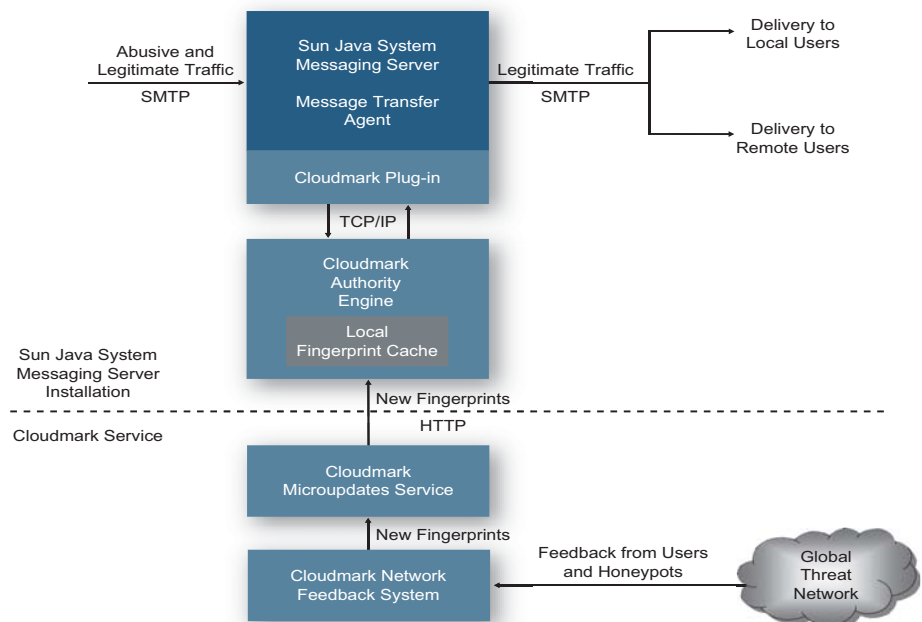


Figure 1. Sun Java System Messaging Server and Cloudmark Authority logical architecture

The Cloudmark Message Filtering Mechanism

Incoming email messages are evaluated using eight unique Cloudmark fingerprinting algorithms to determine the classification of each message. Designed to examine the message for unique characteristics that identify similar spam, virus, or phishing messages, the algorithms are lightweight enough to run thousands of times per second for multiple incoming messages. Each algorithm produces a set of fingerprints for the message in the form of a short hash. These fingerprints are compared against the local cached copy of known abusive fingerprints downloaded from the Cloudmark Micro-Updates service. If any fingerprints match, the message is considered to be abusive, and is classified according to the message type listed in the cache — either spam, phishing, or virus — and given a score which is returned to the Sun Java System Messaging Server. Since a given message generates a fixed set of fingerprints, the time needed to analyze a message is finite, rather than fluctuating according to custom or arbitrary rule sets utilized by other email security systems.

The fingerprinting algorithms are designed for *multiplicity*, attempting to remove the randomness from abusive messages, and detecting the essence of what makes them abusive. As a result, a single fingerprint can match multiple abusive message attacks. Since new abusive messages are often variations or mutations of previous messages, fingerprints of messages already ascertained to be abuse can detect new abusive messages, eliminating the need for users to report the new message. Such multiplicity is the key to the success of the service.

Cloudmark's Automated Abuse Tracking System

Cloudmark's Global Threat Network, which includes users of the Sun Java System Messaging Server system, provides feedback on incoming messages considered to be abuse (false negative) or which are thought to be incorrectly classified as abuse (false positive). This feedback is sent immediately to the Cloudmark Micro-Updates Service, which hosts the Cloudmark Network Feedback System. Each user is assigned a trust rating which improves as they correctly report abuse in corroboration with other, more trusted, users. A user's trust rating drops after incorrectly reporting abuse that disagrees with the majority of more trusted users. If enough trusted users report a message to be spam, virus, or phishing, the fingerprints for that message are sent to the Cloudmark Micro-Updates server for distribution to Cloudmark Authority installations.

Cloudmark Micro-Updates Service

The Cloudmark Micro-Updates service contains a repository of all known abusive messaging fingerprints stored on disk. By default, the Cloudmark Authority Engine contacts the Cloudmark Micro-Updates Service once every minute to retrieve any new

fingerprints. These fingerprints are loaded into memory, thus maintaining an accurate, up-to-date fingerprint cache to evaluate new incoming messages.

On startup and every six hours thereafter, the Cloudmark Authority Engine receives a complete set of fingerprints from the Cloudmark Micro-Updates Service, which it stores on disk. The disk copy is used whenever the Cloudmark Authority Engine restarts so it may process messages immediately, eliminating the requirement for an immediate connection to the Cloudmark Micro-Updates service.

The Sun Java System Messaging Server and Cloudmark Network Architecture

Figure 2 shows a simplified view of the load balancing provided by the Sun and Cloudmark solution. The figure illustrates the use of load balancers between the Sun Java System Messaging Server MTA and the Cloudmark Authority Engine to achieve optimum availability.

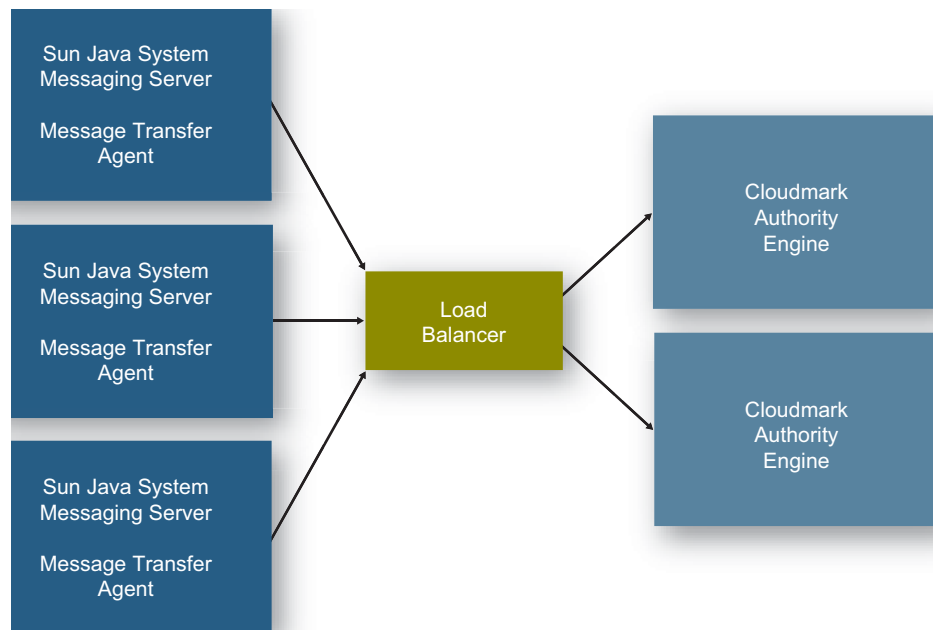


Figure 2. Load Balancing between Sun Java System Messaging Server and Cloudmark Authority Engine

- *Service separation*

The Cloudmark Authority Engine and the Sun Java System Messaging Server system are typically deployed on separate servers to provide for independent capacity expansion, as well as improved availability. It is possible, however, to deploy the Cloudmark Authority Engine on the same host as the Sun Java System Messaging Server MTA.

- *Load balancing*

The Cloudmark Authority Engine servers typically sit behind a load balancer which automatically distributes requests from the Sun Java System Messaging Server system between the Cloudmark servers, and handles any availability issues between the two sets of servers.

- *Security*

According to firewall rules, the Cloudmark Authority Engine servers are not directly accessible from the Internet or customer-facing networks. Usually an outbound rule is created in the firewall to allow the Cloudmark Authority Engine servers to contact the Cloudmark Micro-Updates service. For complete security, the servers can be placed on a separate network using a secure proxy to access the Cloudmark Micro-Updates Service.

- *Redundancy*

Cloudmark Authority Engine servers are typically deployed in a highly available N+1 configuration, so that the loss of a single server does not affect the operation of the service in any way.

Figure 3 shows a typical physical network architecture with all aspects of the system deployed for maximum availability. The configuration relies on the ability of load balancers to balance traffic to and from the same subnet.

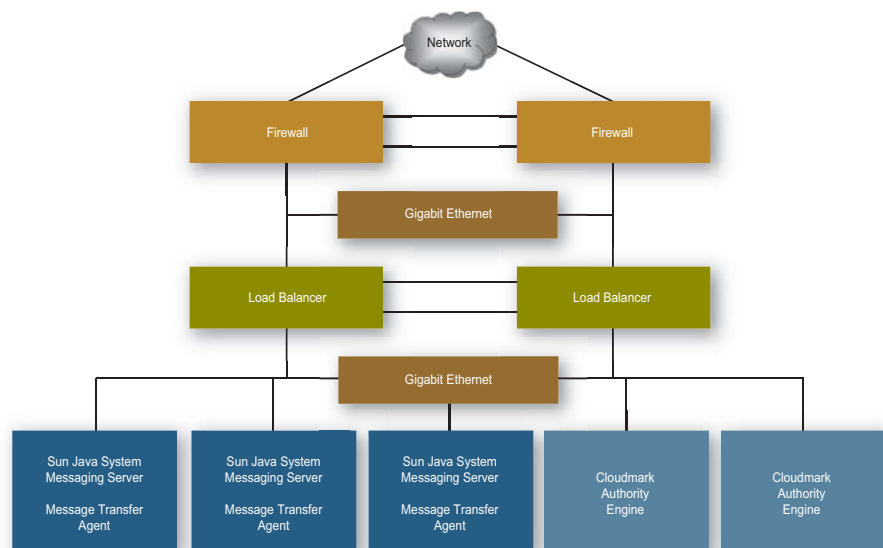


Figure 3. Typical Sun Java System Messaging Server and Cloudmark Authority Network Architecture

Handle More Subscribers, Gain Accuracy and Performance

The unique design of the Cloudmark Authority software addresses the key requirements of service providers for an enterprise-class, carrier-grade messaging security solution. Indeed, the scalability, availability, and performance characteristics needed by large-scale solutions are provided by filtering, detection, feedback, and update functions, along with the tiered architecture that powers it. Innovative design features actually transform the solution, improving accuracy and performance as the subscriber base grows. The mechanisms of that transformation are discussed below.

Greater Filtering Engine Scalability

The filtering component of the Cloudmark Authority software is an extremely high performance engine with the following characteristics:

- *Vertical scalability*
Fully multi-threaded for maximum scalability on servers with multiple CPUs, the engine takes full advantage of the processing power of servers with four to eight CPUs.
- *Horizontal scalability*
The Cloudmark Authority Engine software uses a local in-memory cache of all fingerprints detected in the previous seven days. This approach provides excellent coverage for filtering purposes, since each fingerprint is tiny and the total memory cache is also quite small. As a result, the cache fits easily into the memory of even the most lightweight servers. Each filtering engine server can keep a full copy of the fingerprints in memory, so the solution does not suffer when deployed horizontally. In contrast, updates in software from other vendors can be very large, making it difficult for servers to keep the entire data set in memory, thus limiting the ability of the software to scale horizontally.
- *Client-Server architecture*
The architecture design makes it possible for Cloudmark Authority Engine Servers to be deployed on hardware separately from the Sun Java System Messaging Server MTA and scaled appropriately. Separate deployment makes it easy to expand capacity exactly where it is needed, enhancing the scalability of the solution.

Scaling the Feedback and Update Architecture

The Cloudmark CNFS and update architecture is designed specifically to scale to handle millions of subscribers and billions of messages. To achieve that scalability, the filtering system must be capable of operating without manual processing. Many other systems require human intervention as an essential element in the day-to-day operation of the system, receiving feedback from honeypots, users or both, and creating, testing, and deploying rules to address the feedback. Such a system works well for a small to medium set of subscribers and low to medium message volumes, but can not scale and

quickly becomes unmanageable as the number of subscribers and messages grows. Processing the message filtering feedback requires a sizeable staff, slows the solution with growing updates, and fosters human error, potentially causing catastrophic performance and availability problems.

In contrast, Cloudmark designs the service to operate with minimal manual intervention. The automated feedback, trust evaluation and feedback mechanism, combined with unique and innovative fingerprinting algorithms, enables the system to not only work well in an automated manner, but actually perform even better and faster with increasing numbers of users.

Cloudmark recognizes the skilled human element is still critical, and employs an expert research department to closely examine system operation, fine-tune the service, and implement strategic initiatives such as developing new algorithms to fight new threats, such as advanced phishing and image spam. By harnessing the horsepower of Sun systems to automate the feedback and update process, and leveraging the intelligence of Cloudmark's research department to continually improve the system, the Cloudmark and Sun solution is able to scale and combat messaging abuse like no other offering.

More Performance

The predictable performance and high speed of the Cloudmark Authority Engine can be attributed to several factors.

Extremely Efficient Algorithms

The Cloudmark Authority Engine uses eight algorithms to generate fingerprints for an incoming message, which are checked against known *bad* fingerprints. The time needed to generate a set of fingerprints is therefore bounded. This approach differs from most solutions which have hundreds, if not thousands, of manually generated rules, each of which must be checked before a message can be classified correctly. This process consumes time and computing resources, resulting in unpredictable performance that is typically slow.

Fast Hash-Based Cache Lookups

Once a set of fingerprints is generated for an incoming message, it is checked against a cache of *bad* fingerprints stored in memory. The fingerprints are generated in the form of short hashes, so the comparison can be achieved using extremely fast hash-based exact matching algorithms. Additional optimizations, such as checking each fingerprint before the next one is generated, can be used to further increase performance.

Optimal Algorithm Coding

By utilizing a relatively fixed set of algorithms, the Cloudmark research team can spend more time optimizing algorithm performance before deployment, and avoid poor

designs which could adversely affect performance. By comparison, many competing solutions rely on rules which are manually created in response to live threats and suffer from performance problems due to lack of testing and quality assurance.

Speedy Attack Response

The automated nature of the Cloudmark Micro-Updates Service enables new threats to be identified and blocked within a few minutes of the attack first being detected. In addition, since the fingerprints for messages are designed for multiplicity, variations on previously staged attacks can be blocked without the need for feedback on those attacks. The need for a speedy response to attacks can not be overemphasized. By using Cloudmark technology, recent phishing attacks seen by Cloudmark typically last only a few hours before the attack is dismantled, compared to days or weeks when service providers use competitive solutions.

Authority Performance Statistics

Table 1 lists performance figures for the Cloudmark Authority Engine on several different types and generations of Sun hardware, as performed under lab conditions. These results reveal the Cloudmark Authority Engine performs extremely well on current or previous generation Sun systems.

Table 1. Cloudmark Authority Engine performance statistics

Server	Configuration	Sustained Peak @75 Percent CPU Utilization Messages/Second	Average Message Size
Sun Fire T1000	6 x 1 GHz cores 4 GB RAM	1137	10 KB
Sun Fire T1000	6 x 1 GHz cores 4 GB RAM	890	27 KB
Sun Fire T2000	8 x 1.2 GHz cores 16 GB RAM	402	50 KB
Sun Fire V480	4 x 1.2 GHz cores 16 GB RAM	368	10 KB
Sun Fire T2000	4 x 1.2 GHz cores 16 GB RAM	253	25 KB

Highly Available

Cloudmark Authority is designed from the ground-up to provide a resilient, highly available service. Key availability features of the Cloudmark Authority Engine make it capable of fulfilling the most demanding service provider SLAs.

Stateless Architecture

Cloudmark Authority is by its nature a stateless service — no information is stored about transactions, and each transaction takes place independently of any other

transaction. A stateless architecture makes it possible to retry a request on another server with no loss of data or context if the server is lost due to hardware or software failure.

Resistant to Failures

Cloudmark Authority uses a cache of bad fingerprints to evaluate messages. The cache is stored in memory for performance reasons, but can be retrieved from disk if the Cloudmark server is restarted. Any updates which are not contained in the on-disk copy are downloaded from the Cloudmark Micro-Updates service after startup, and complete copies of the fingerprint cache are downloaded and saved to disk every six hours. The software also maintains several backup copies of the cache file to guard against data corruption.

In the event of a failure, the Cloudmark Authority Engine continues to evaluate messages using its cached copy of fingerprints. The cache is complete, with the exception of updates generated after the failure, so the filtering continues quite effectively. In this manner, the architecture can handle transient network failures, server failure, loss of the most recent saved cache, or any combination of the above events.

Resilient Client-Server Architecture

The integration of Sun Java System Messaging Server and Cloudmark Authority software uses a client-server architecture, so all requests from the Sun Java System Messaging Server MTA to the Cloudmark Authority Engine pass over a TCP/IP connection. This tiered approach results in the following availability enhancements:

- The Cloudmark Authority Engine does not need to reside in the address space of the Sun Java System Messaging Server MTA. As a result, a crash or problem in one service does not affect the operation of the other service.
- Requests can be directed at multiple servers, utilizing server redundancy along with techniques such as load balancing.

Load Balancing to Increase Availability

One of the most effective mechanisms to spread load between servers and protect against network, hardware, and software failures is to use hardware load balancers. Load balancers are in use at many of the world's most highly available and scalable infrastructure sites to not only detect and route around failure, but also ensure an automated mechanism for large-scale, efficient use of resources.

The protocol between the Cloudmark Plugin in the Sun Java System Messaging Server MTA and the Cloudmark Authority Engine software enables load balancers to be used without any special configuration. Requests are made against the virtual address, and the load balancer uses appropriate algorithms to direct the request to the most

appropriate Cloudmark Authority Engine server. Requests that fail are immediately retried, in which case the load balancer redirects the request to an available server.

Handling Failure Conditions

Table 2 shows the types of failures that may occur in a service provider installation of Sun Java System Messaging Server and Cloudmark Authority software, along with details of how the Cloudmark Authority software copes with each failure condition.

Table 2. Types of failure and coping mechanisms

Failure Condition	Failure Handled?	Details
Authority Engine server failure (hardware, software)	Y	Load balancing redirects requests to other servers Redundancy ensures spare capacity
Network failure between Sun Java System Messaging Server installation and Cloudmark Micro-Updates Service	Y	Cloudmark Micro-Updates Services located in multiple Network Operations Centers (NOCs), geographically distributed Cached copies of fingerprints enable service to continue to filter abusive messages efficiently
Corrupted cache file on disk	Y	Multiple versions of file are stored on disk, and older versions can be used (requires operator intervention) Complete new cache files are downloaded every six hours In cases of disk read failure, the service still attempts to download a new full set of cache files
Transient failure of Authority Engine (e.g. network problems between Sun Java System Messaging Server MTA and Authority Engine server)	Y	Cloudmark Plugin retries connection, and load balancer redirects to available server
Load balancer failure	Y	Deploy redundant load balancers
Cloudmark Authority Engine servers running low on capacity	Y	Deploy new Cloudmark Authority Engine servers and introduce seamlessly into service using load balancer

Sun and Cloudmark

The integration of the Cloudmark Authority software and Sun Java System Messaging Server provides a carrier-class anti-abuse solution that fulfills key service provider requirements for scalability, performance, and availability. The Sun and Cloudmark solution scales to the highest volumes of message throughput, with world-record performance and reduced latency. Deploying Cloudmark software on a Sun platform can lead to demonstrable savings in capital and operational expenditures, with a clear ROI advantage over the competition. With industry-leading availability features to

guard against a wide range of failure conditions and resilience built in to the design, the Sun and Cloudmark solution meets the demands of the largest and most successful providers of messaging services.

Cloudmark is a global leader in carrier-grade messaging security, delivering the industry's fastest, most comprehensive and most accurate real-time spam, virus, and phishing protection. Cloudmark solutions are currently protecting more than 180 million mailboxes and over 80 of the world's largest service providers and mobile operator networks have replaced competing solutions with Cloudmark. Serving on the boards of the Messaging Anti-Abuse Working Group and Anti-Phishing Working Group, Cloudmark is a leader in the messaging industry.

For over two decades, Sun has continued to provide flexible, scalable, innovative, and cost-effective solution infrastructures. Incorporating the latest technologies, next generation processors, and improved reliability, manageability and serviceability features, Sun solutions facilitate the adoption of leading edge, high-bandwidth infrastructures, service-oriented architectures, and services. Together, Cloudmark and Sun offer a powerful solution that helps service providers maintain a safer environment free from messaging abuse.

For More Information

For more information on Sun and Cloudmark products and technologies, contact a Sun sales representative or visit the Web sites listed in Table 3.

Table 3. Sun and Cloudmark Web sites for further information

Web Site URL	Description
cloudmark.com/serviceproviders/	Cloudmark Products
sun.com/servers	Sun Servers
sun.com/servers/coolthreads/t1000	Sun Fire T1000 Server
sun.com/servers/coolthreads/t2000	Sun Fire T2000 Server
sun.com/servers/coolthreads/tnb	Sun Fire CoolThreads Try and Buy Resources
sun.com/solaris	Solaris™ Operating System
sun.com/tryandbuy	Sun Try and Buy Program
sun.com/security	Security
sun.com/jes	Sun Java Enterprise System and Sun Java System Suites
sun.com/software/products/messaging_srvr/index.xml	Sun Java System Messaging Server

