



Profense™ 2.2

White-paper

January, 2008

Armorlogic, the Armorlogic logo, the Armorlogic logotype and Profense™ are trademarks of Armorlogic ApS. Products mentioned herein are for identification purposes only and may be registered trademarks of their respective companies. Specification subject to change without notice.

Copyright © 2008 Armorlogic ApS

Table of Contents

A proactive approach to web security	1
A typical website of today	2
Web applications	2
The web security challenge	3
Reactive protection methods are dominant	3
Securing legacy systems is costly	4
Profense™ - a proactive scalable approach	5
Move from reactive patching to proactive protection	5
Secure applications without reprogramming	5
Scale performance transparently	5
Business benefits of Profense™	6
Reduced risk	6
Cost reduction	6
Efficiency improvement	7
Profense™	8
Load Balancer	9
Session persistence	9
Guaranteed secure persistence	9
HTTP and HTTPS request switching	9
Web Accelerator	11
Caching	11
HTTP compression	11
SSL acceleration	11
TCP connection off-loading	11
Web Application Firewall	12
Automated application profiling	12
Adaptive learning with instant protection	12
Positive security model	12
Negative security model	13
Proactive protection	13
Filtering	14
Management	15
Log functions	16
The Profense™ Platform	17
Platform features	17
Platform technology	18

A proactive approach to web security

Web sites are becoming increasingly important for companies and organizations. Crucial information and business processes are managed by Web sites, so that their reliability, usability and overall quality are central issues. Web security is therefore on the management agenda of most public and private organizations as these issues are closely related to it- security.

Organizations of today require robust, reliable and scalable web sites. They realize that however attractive the Web is as a channel to deliver goods and services to customers and citizens, without confidence that private information is being protected, the potential of the Web will not be realised.

Also legislation and regulation like Sarbanes Oxley, Payment Card Industry Data Security Standard, the Gramm-Leach-Bliley Act, the Data Protection Act, etc. are requiring that controls are in place to ensure protection of data.

Protecting a web site from hacker and worm attacks and from long response times due to overload is a complex and resource demanding ongoing task though. This is because a typical dynamic web site is constantly changing, content and software are updated and new vulnerabilities in standard software are disclosed requiring patches to be applied.

A typical website of today

During the past decade the average website have developed into a fairly complex collection of web applications and data sources.

Former

A collection of static pages and graphics and simple usage of CGI scripts with simple database access, typically only to one source.

The average website of today

Complex applications in several layers written in C#, Java, Perl, PHP, .NET or other languages with access to a number of data sources.

A collection of *web applications* with complex business logic.

Web applications

Web applications are basically software programs which are accessible from the Internet. They play a major part in the overall security of a web site. Even though network firewalls are installed, off-the-shelf software is patched and communication is protected with heavy encryption, there are many ways to attack the logic of the custom-made application code itself. Web applications often access critical data sources and internal systems and are therefore the prime target for more serious attacks.

In order to avoid security incidents web applications should therefore (of course) be developed in keeping with best practises for secure programming. Often though - they are not. It is our experience that 8 out of 10 websites have problems with for instance input validation in web applications leading to vulnerability to attacks which compromises confidentiality and integrity of business critical data.

The use of web applications is widespread ranging from simple CMS systems to complex e-government, B2B or B2C applications with integration to back-end databases, ERP or CRM systems. Most websites of today uses web applications.

Due to complexity and constant change web applications and web systems in general are weak spots in many organizations internet perimeter. They represent an ongoing resource demanding challenge.

The web security challenge

For most organizations a web attack can involve significant loss and brand damage. They therefore go to great lengths to protect the website from such incidents. But a number of factors make the task a resource demanding never ending task with limited success.

No two web applications are the same

Most web applications are custom built or altered versions of standard software. This means that every organization have go through the process of finding vulnerabilities and developing and applying patches to their web applications. Unlike with standard software, no others are trying to find potential vulnerabilities and no patches are available for download.

The process of finding and fixing web application vulnerabilities is time consuming and resource demanding.

Frequent discovery of new vulnerabilities

New exploits are becoming available faster than a patch is available and can be tested and installed.

Not all vulnerabilities are necessarily disclosed to the public. This is a known fact. Exploits targeting such vulnerabilities, so-called "private exploits" are exchanged or traded in underground communities.

Web systems are complex

A web system often is a heterogenous layered collection of servers, applications and data which requires different means of protection and different skills to operate and secure.

Complexity increases the risk of human error and "admin mistake" is very common reason for an organizations website being hacked.

Reactive protection methods are dominant

Most of the methods in the traditional "security toolbox" are reactive in nature and fail to provide sufficient protection leaving organizations exposed to unnessecary risk.

Vulnerability patching

However important vulnerability patching is, it is a reactive security process which only protects the organizations web systems from publicly known vulnerabilities.

In the time span from a vulnerability is disclosed, a patch is developed and is actually applied, critical internet exposed systems are vulnerable. This time span is known as "the window of exposure".

Any attacker with the knowledge of the vulnerability or having a working exploit can gain unathorized access to vulnerable systems and data in that time span. Vulnerabilities can be exploited either by worms, viruses or through a targeted attack.

Network Firewalls

The network firewall is (a sophisticated) filter working on the network level. In this context it's primary function is to ensure that only systems meant to be internet exposed are accessible from the internet - eg. web servers.

Traffic to web servers, benign or malicious, is indiscriminately passed through.

Some firewall vendors claim to provide protection on the application level because they are able to inspect the payload of packets and identify attacks based on signatures or by other heuristic methods. However, network firewalls do not aggregate the packets into full requests to determine if each request is valid in the context of a particular application and hence cannot determine if a stream of packets together form an attack exploiting a specific vulnerability or misconfiguration of an application or application server.

Intrusion Detection Systems (IDS)

IDS' are truly reactive. IDS passively monitor network traffic and generate an alert when potentially malicious traffic is detected (based on attack signatures, heuristic methods and other statistic methods). IDS help determine *what happened*.

Intrusion Prevention Systems (IPS)

IPS' are less reactive in the sense that they do their best at identifying and blockcking malicious traffic. IPS rely on detecting the malicious traffic based on attack signatures, heuristic and other methods. They suffer from the same shortcomings as all other signature based systems. They only protect against known attacks and require constant updating of the attack signatures.

IPS are thus vulnerable to zero-day attacks exploiting undisclosed vulnerabilities for which no patches or workarounds are made available.

Clearly there is a need for a more proactive approach to protecting web systemens and applications.

Securing legacy systems is costly

The single most important aspect of securing web applications is strong validation of all input parameters to the application.

If a web application does not validate input properly it is exposing data and backend systems to the risk of losing integrity, confidentiality or availability through eg. SQL and command injection attacks.

Unfortunately, weak or insufficient input validation is very common in todays web applications. Furthermore it is costly to redesign and reprogram web applications to employ strong input validation.

Profense™ - a proactive scalable approach

In order to deal effectively with the growing threat from internet organizations need to focus more on controlling access to business content and less on identifying and denying malicious requests from the internet. Obviously the latter objective is achieved through the first but not vice versa. That is: basing it-security on the assumption that everything not explicitly allowed is per definition forbidden.

This approach is a more durable less resource demanding means of protecting web systems. It protects against attacks targeting undisclosed vulnerabilities in standard software and it defeats directed attacks from professional it-criminals targeting weaknesses in the organizations own custom built web applications. Simply because the protection is based on modelling acceptable input.

Based on a positive security model, Profense™ supports this approach. It provides true proactive protection of web servers and web applications by only accepting legitimate requests to the organizations web systems and applications. Anything else is per definition malicious and is rejected.

Move from reactive patching to proactive protection

With Profense™ the organization move from reactive patching to proactive protection. The protection is substantially enhanced and It provides your organization with a time buffer to appropriately test, configure and deploy patches for new vulnerabilities in a more planned fashion and thereby reducing costs associated with emergency patching, short cutting change management procedures and system cleanup.

Secure applications without reprogramming

Profense™ allows for the Implementation of strong positive protection of web applications in a fraction of the time it would take to protect the same application by reprogramming it - without access to the source code and without programming skills.

Scale performance transparently

A load balancing module allows for transparent scaling of web site performance by adding extra web servers when needed. It is not a problem if the web site is stateful in the sense that once a visitor is directed to a web server it is desirable that all subsequent requests from that visitor is directed to that same web server. This is handled transparently by enabling *session persistency* in Profense™.

Business benefits of Profense™

For e-enabled organizations Profense™ maximize business value of web presence by reducing traffic cost by 30-60%, improving efficiency of content delivery and reducing business risk.

Profense™ offer lower TCO because it is easier to manage and configure, it's more scalable, the licensing model is granular and the costs of disaster preparedness and recovery are significantly lower than similar appliance based solutions.

Reduced risk

Profense™ assures uninterrupted availability, integrity and confidentiality of systems and data by:

- Distributing and balancing load between available servers, avoiding overburdened or unavailable servers.
- Proactively protecting against attacks from hackers and worms by employing a positive security model which even protects against attacks targeting unpublished or unpatched vulnerabilities.

Cost reduction

Profense™ reduce operational cost by:

- Saving 30-60% on bandwidth usage by compressing traffic.
- By reducing workload and balancing load between web servers thus reducing the need for extra web servers.
- Freeing administrative resources by releasing the constant burden of web server patch management.
- Providing a cost efficient alternative to securing legacy web applications.

Lower cost of performance scaling

Profense™ runs on general purpose server hardware and the licensing model is not based on hardware performance resulting in significantly lower cost of scaling performance.

- Improve performance of a single node Profense™ by upgrading hardware with no extra license cost.
- Improve performance of a Profense™ cluster by adding nodes or upgrading hardware with no extra license cost.

Lower cost of disaster preparedness and recovery

If disaster strikes all that is needed to restore Profense™ is a backup and a piece of general purpose server hardware.

For competitors appliance based solutions a new expensive specialized piece of hardware have to be procured or kept standby off-site.

Efficiency improvement

The load balancing and web acceleration modules improve system efficiency by:

- Speeding up content delivery
- Utilizing system resources and bandwidth more efficiently

Profense™

The Profense™ platform is a modular web application assurance system offering acceleration, scalability and proactive protection of web systems. The following modules are available:

Load Balancer

Enabling scalability and acceleration of even complex SSL-enabled stateful web applications.

Web Accelerator

Reducing traffic cost, improving response time and off-loading web servers.

Web Application Firewall

Proactive protection of web servers and web applications by employing a positive security model.

Load Balancer

The Profense™ Load Balancer module enables scalability and acceleration of even complex SSL-enabled web applications.

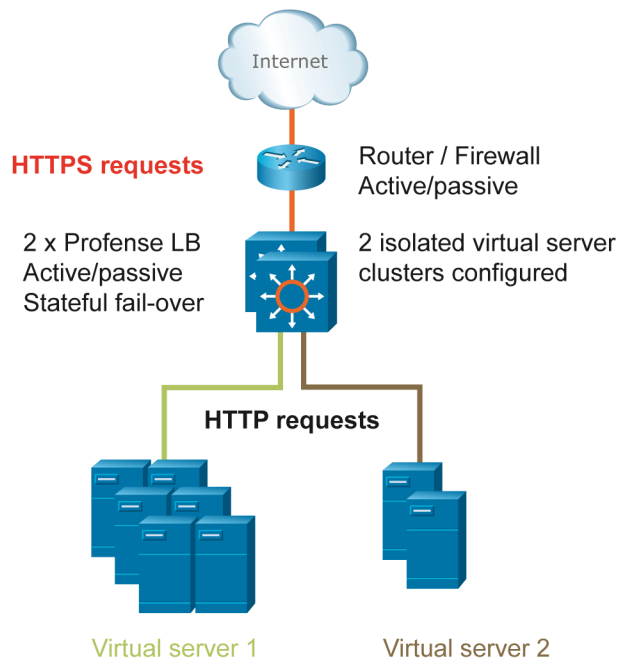


Figure 1. Load balancing HTTPS

Session persistence

Session persistence is achieved through insertion of a cookie tracking the session.

When the Profense™ Load Balancer is configured in an active/active cluster (is load balanced itself) the session persistence is independent of the cluster node handling the request.

Guaranteed secure persistence

Profense™ Load Balancer offers guaranteed SSL session persistence by decrypting the SSL content.

In this way the client is guaranteed a secure persistent browsing experience without loss of state information.

HTTP and HTTPS request switching

As SSL-connections are terminated by Profense™ it works equally well with HTTP and HTTPS.

Optionally requests from clients can be re-encrypted before being forwarding to back-end servers.

Requests are distributed to backend servers following one of the two methods below:

Round robin

Requests are distributed equally in a round robin fashion to all active servers.

Session persistence

When a server is selected according to the methods above all subsequent requests for the same client can be sent to the same physical server in order keep state information for that client on that server. This method is also referred to as client stickyness.

Web Accelerator

The web accelerator module accelerates web application and web system performance by:

- Lowering the web and application server workload
- Optimizing and reducing bandwidth usage
- Offloading SSL operations from web servers
- Optimizing TCP-connection handling

Caching

Caching of static documents improves performance by 300 - 500%.

Documents that can be cached, are locally stored by Profense™. Any further requests for documents found in the cache, are automatically delivered to clients directly from Profense™. Therefore, the back-end web servers can focus on delivering dynamic content with improved response times to clients, without the overhead of delivering static content like images, PDF documents, static HTML documents, style-sheets and others.

HTTP compression

Dynamic compression of transmission data reduces bandwidth usage by 30 - 60% and increases transfer rate by 50 - 100%.

HTTP compression reduces the transfer volume of static and dynamically generated web pages to approximately 1/3 of their original size proportionally speeds up the load time performance. This results in reduced traffic costs and in a better experience for the web site visitors.

SSL acceleration

Profense™ has the ability to terminate HTTPS (SSL) based connections and requests from clients before forwarding them as HTTP non-SSL) to back-end servers.

This off-loads the back-end web servers from expensive SSL calculations thus allowing them to focus on faster content delivery to clients.

TCP connection off-loading

When forwarding legitimate requests from clients to back-end web servers, Profense™ will reuse socket connections already established with the back-end web server.

This gives a performance increase since back-end servers don't waste resources on establishing new and tearing down old socket connections.

Web Application Firewall

Profense™ Web Application Firewall is implemented in the network as a filtering gateway which validates all requests to the web systems.

On a general level the web application firewall module has the following protective features:

- Web server cloaking and customizable HTTP error handling completely shield web servers from direct Internet access and defeat fingerprinting attacks.
- White-list based filtering of input data (including all URLs and parameters) allows for protection against threats from unpublished vulnerabilities in web server software and web applications.
- Validation of requests using a combination of positive and negative policy rules. Available in Profense Professional.
- HTTPS termination allows for white-list based protection from SSL-encrypted attacks.
- The protection is always updated as there is no dependence on signatures due to the positive security model.

Automated application profiling

All versions of Profense™ includes the automated application profiling, or learning, engine which allows for completely automated policy building.

Adaptive learning with instant protection

Profense Professional offers `Auto mode` using a combination of positive and negative policy rules with adaptive learning of changes in the web applications. The Auto mode provides instant protection which improves as Profense learns the web applications and consequently can create positive policy rules for critical application components.

Positive security model

Profense™ is based on the positive security model. It determines allowable requests, and inputs and disallows everything else. This approach provides protection against unknown threats, simply because they are not in the white-list and thus are disallowed.

The working basis of the positive security model is that everything is forbidden unless explicitly allowed. In the context of Profense™ this implies that only allowed requests are forwarded to the web system - that is: requests for web pages, applications, parameters etc. which you allow. This positive security approach is proactive because you base your protection on known information, the business content you want your web system to make available, not attack signatures and other potentially unknown information.

Negative security model

In Profense™ Professional the negative security model - signatures matching known attacks - can be used in combination with positive policy rules. For example it is possible to specify (or learn) strict positive input validation rules for certain critical application components, like login.php or payment.jsp, and use more general negative signatures for the remaining part of the web site.

Proactive protection

Because of Profense™'s positive security model it stops exploits of vulnerabilities and weaknesses without dependence on signatures. By building an access control list based on a finite amount of information, the business content of the web system, Profense™ effectively blocks attacks from hackers and worms.

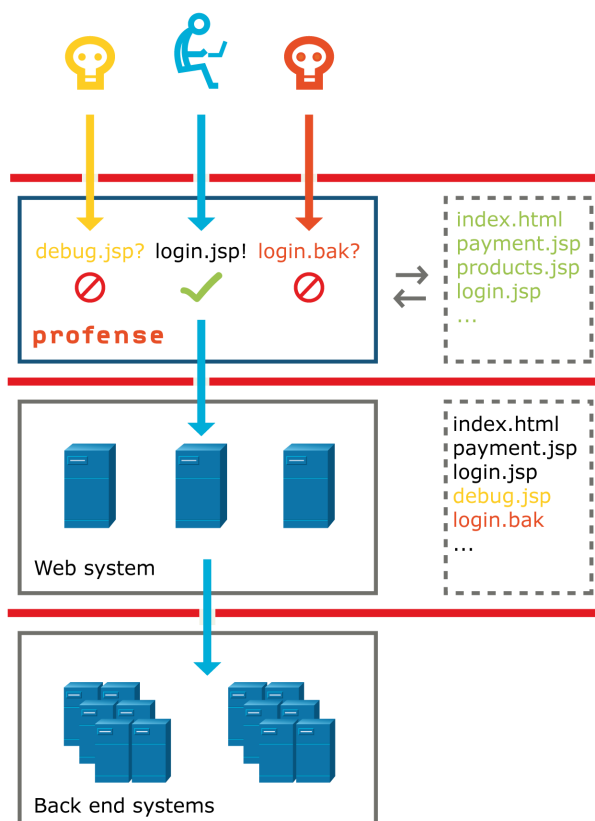


Figure 2. Positive filtering in a web context

In other words: Profense™ does not identify attacks, it determines if a request is allowed based on a white-list. If a request is not in the list it is treated as if it was an attack. This means that Profense™ also protects from attacks targeting unpublished vulnerabilities.

Filtering

Positive and negative URL filtering

Profense™ validates all parts of a HTTP request (including the path, query and segment) according to the defined access policy.

Requests not-matching the access-policy, are per default flagged as illegitimate, rejected and logged for further analysis. This allows system administrators to have a strict white-list of legitimate URLs for a given web application.

In Profense™ Professional policy rules can be specified using positive matching for specific URL and negative for all other.

Positive and negative query filtering

Profense™ validates all parts of a query in a URL request according the defined white-list access policy.

Each parameter and the corresponding value is validated. This allows system administrators to specify what input is allowed for a given URL resource.

In Profense™ Professional, as for URLs, combinations of negative and positive policy rules can be employed.

Global parameter wild-cards

Rules which match parameters on a global basis can be specified using regular expressions or signature based matching (in Profense™ Professional).

This is particularly useful when for instance the web application uses global parameters for session tracking or for printer friendly displaying instructions.

HTTP headers compliance checking

Profense™ can enforce pragmatic and strict standard HTTP headers compliance (RFC2068/ RFC2616).

All request from clients are validated against a strict positive list of valid HTTP headers and values. This prevents possible attacks that exploit vulnerable web applications and servers through illegal HTTP headers.

Strict HTTP headers compliance checking can potentially cause problems with clients that deviate from standards or are otherwise incompatible.

Pragmatic HTTP headers compliance checking is a more loose access policy enforcement comparing to the strict method described previously which still protects web applications and servers from validating values for the submitted parameters according to a positive list of allowed data compiled from the strict RFC compliant heads.

Pragmatic HTTP headers checking allow non-standard headers to pass through Profense™.

Web server cloaking and isolation

Profense™ completely isolates the web server from direct Internet requests and information and web system technology information is removed from web server responses.

A typical web server gives out a lot of information about its version, installed software, operating system, etc. This information is completely irrelevant for normal HTTP/HTTPS communication between clients and web server. However, attackers and worms can misuse this information to craft more targeted attacks on a vulnerable web application or server. Profense™ removes this information from the response sent back from the back-end server before forwarding it to the client thus protecting the web application and server from leaking potentially sensitive information.

Profense™ terminates all HTTP/HTTPS requests from clients before forwarding legitimate requests to back-end web servers. This means that back-end web servers are isolated from clients (typically from the Internet) and are placed on a back-end network/LAN segment. Network isolation means that only HTTP/HTTPS traffic is actually forwarded to the back-end servers. Any other network traffic (for instance a ICMP flood attack or a request to another potentially vulnerable service) will never reach the back-end servers thus eliminating other network threats as well.

Management

Automated Policy Generation

Profense™ automatically generates access policies for even complex web applications and web systems.

All relevant information for a web application including URLs, parameters and HTTP methods is automatically learned by Profense™ and applied to the running access policy. This allows system administrators to quickly enable new or updated information about the web application thus reducing the manual work needed when implementing new or changed access policies.

Regular expressions support

Profense™ has full support for standard PCRE (Perl Compatible Regular Expressions).

This feature allows system administrators to manually fine-tune and implement strict values for legitimate HTTP parameters.

Global URL wild-cards

In order to simplify the ACL Profense™ supports the definition of URL wild cards based on regular expressions which matches URLs without parameters on a proxy global basis.

Global parameter wild-cards

Rules which match parameters on a global basis can be specified using regular expressions.

This is particularly useful when for instance the web application uses global parameters for session tracking or for printer friendly displaying instructions.

Class based policy rules

Filtering rules can be specified using classes for easy administration.

Classes are defined globally and can be applied both when manually editing the access policy, when the access policy is built automatically and when rules are added or modified from log.

Log functions

Attack classification

All rejected requests are classified in major attack groups (i.e. SQL-injection, buffer overflow, etc.) using a combination of cross validation, heuristic patterns and statistics.

External notification

Alerts can be sent to external syslog server or email.

Alert levels are completely configurable and are mapped to standard syslog priorities (information levels).

Detailed logging

The management interface includes a comprehensive security log displaying all the necessary details about blocked requests, including the time stamp, IP address, HTTP methods, path and query segments, HTTP headers violations, attack classification and raw request data.

Customizable search criteria

Multiple search criteria can be specified using wildcards allowing for detailed drill down searches. Customizable reporting.

Customizable reporting

All log views (search filter sets) can be exported to printable reports or XML.

Access log

In Profense™ Professional all requests to the website can be logged in addition to the deny log..

The Profense™ Platform

Platform features

Ease of use

Though ease of use is a qualitative statement we prefer regarding it as a product feature in order to keep development focus on the usability aspect. Misconfiguration of systems is a major source of vulnerability. Therefore ease of use is a security feature as it increases the likelihood of getting things right the first time and thus reduces the risk of human error due to complexity and misconceptions.

Ease of use is achieved through:

Simplifying the complex

A fundamental concept of Profense™ is simplicity. There are no bells and whistles and every feature reflects real life needs of security demanding organizations.

Clear and intuitive graphical user interface

The web based graphical user interface provides access to manage all system and module functions.

Operation

Automated remote backup

The complete running Profense™ installation including all settings, proxies and access policies can be automatically backed up by Profense™ to a remote FTP server.

This feature is available in Profense™ Professional.

Manual full and partial backup

A complete Profense™ installation or the entire configuration of a single proxy can also be backed up manually with a few clicks in the management interface.

Easy restore

A complete Profense™ configuration including access policy for all defined proxies can be restored from an FTP-server or the file system with a few clicks in the management interface.

Scalability and availability

Clustering

Active/active clustering with automatic policy synchronization allows for virtually unlimited scalability.

This feature is available in Profense™ Professional.

High availability

Profense™ can be run in active/passive configurations where two or more physical Profense™ nodes together comprise a logical Profense™ unit with hot fail-over and automatic synchronization of rules across the units.

Automatic synchronization is available in Profense™ Professional.

Platform technology

Profense™ is based on proven methods and technology.

With Profense™ the organization gets transparent state of the art protection of web systems and web applications - without compromising functionality and software and hardware policies.

Software appliance

Profense™ combines the flexibility and scalability advantages of software with the security advantages and administrative simplicity from dedicated hardware appliances. The Profense™ software appliance installer turns a piece of general purpose application server hardware into a dedicated application acceleration and security gateway within minutes - with minimal interaction.

The Profense™ software package is completely self contained and no system level expert know-how or low level interaction is required to install and run Profense™.

The web based administrative interface provides access to perform all necessary administrative tasks, including initial configuration, administration of clustering and filtering rules, backup, log and reporting functions.

Profense™ OS

Profense™ is based on a stripped and hardened OpenBSD platform which is regarded as the most secure OS generally available.

The proxy, filtering and administration components run in a non-privileged and closed runtime environment and technologies like ProPolice, W^X protection, non-executable stack, etc. are used to further harden the system against attacks.

With Profense™ you get a seriously hardened and secured frontend to your web applications - without compromising functionality.