CHƯƠNG 10
WEB APPLICATION FIREWALL

12/29/2024

ThS.Nguyễn Duy
duyn@uit.edu.vn

Content

- What is WAF?
- Why we need to use WAF?
- Web Application architecture
- How does WAF prevent attacks?
- How to deploy WAF?

Content

- **What is WAF?**
- Why we need to use WAF?
- Web Application architecture
- How does WAF prevent attacks?
- How to deploy WAF?

What is WAF?

- A web application firewall (or WAF) filters, monitors, and blocks HTTP/HTTPS traffic to and from a web application.
- A WAF is differentiated from a regular firewall in that a WAF is able to filter the content of specific web applications while regular firewalls serve as a safety gate between servers.
- By inspecting HTTP traffic, it can prevent attacks stemming from web application security flaws, such as SQL injection, cross-site scripting (XSS), file inclusion, and security misconfigurations

TOP 10 OWASP – 2022

https://owasp.org/www-project-top-ten/

- Injection
- Broken Authentication
- Sensitive Data Exposure
- XML External Entities (XXE)
- Broken Access Control
- Security Misconfiguration
- Cross-Site Scripting XSS
- Insecure Deserialization
- Using Components with Known Vulnerabilities
- Insufficient Logging & Monitoring

Web Attack Damage

- Loss of sensitive data
- Defaced Web site
- Lost Business
  - Web site blocked by search engines and AV software
  - Loss of customer trust

How Widespread Are Application Attacks?

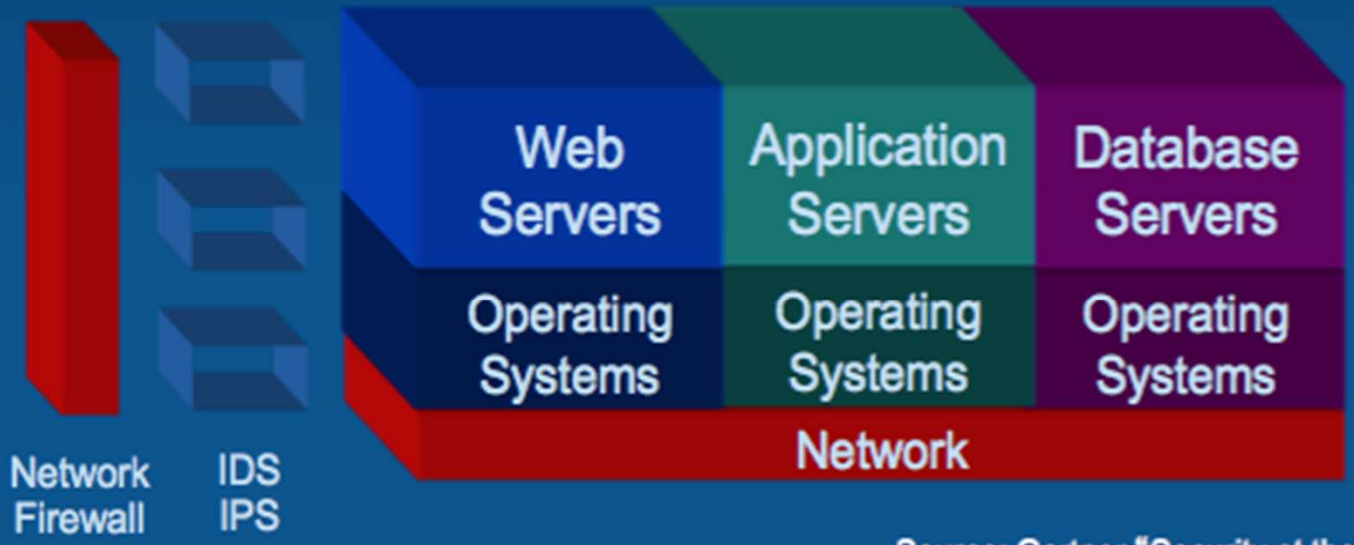
75% of Attacks Focused on the Web applications

SQL Injection
Parameter Tampering
Cross-Site Scripting
Other Attacks

Customized Application Code

- Rushed to Production
- Written Before Security was a Priority

Confidential Data

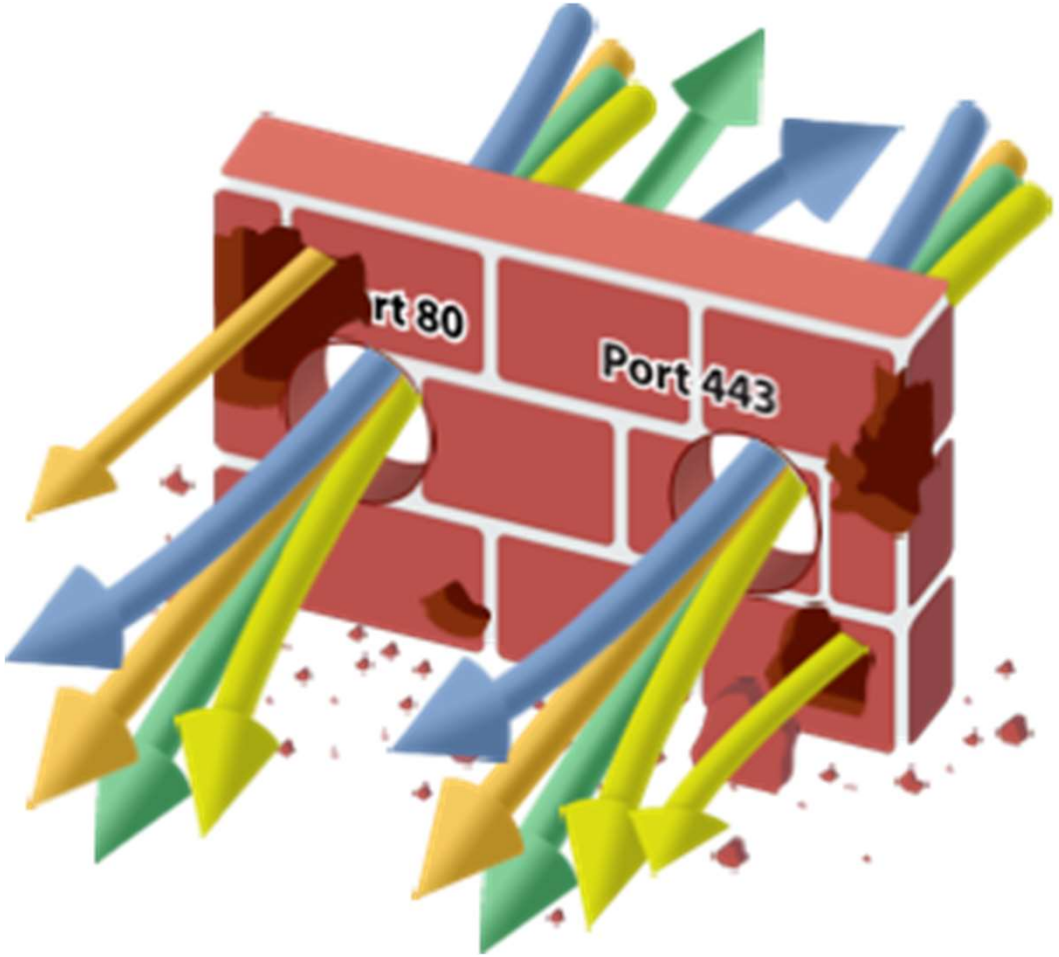


Source: Gartner "Security at the Application Level"

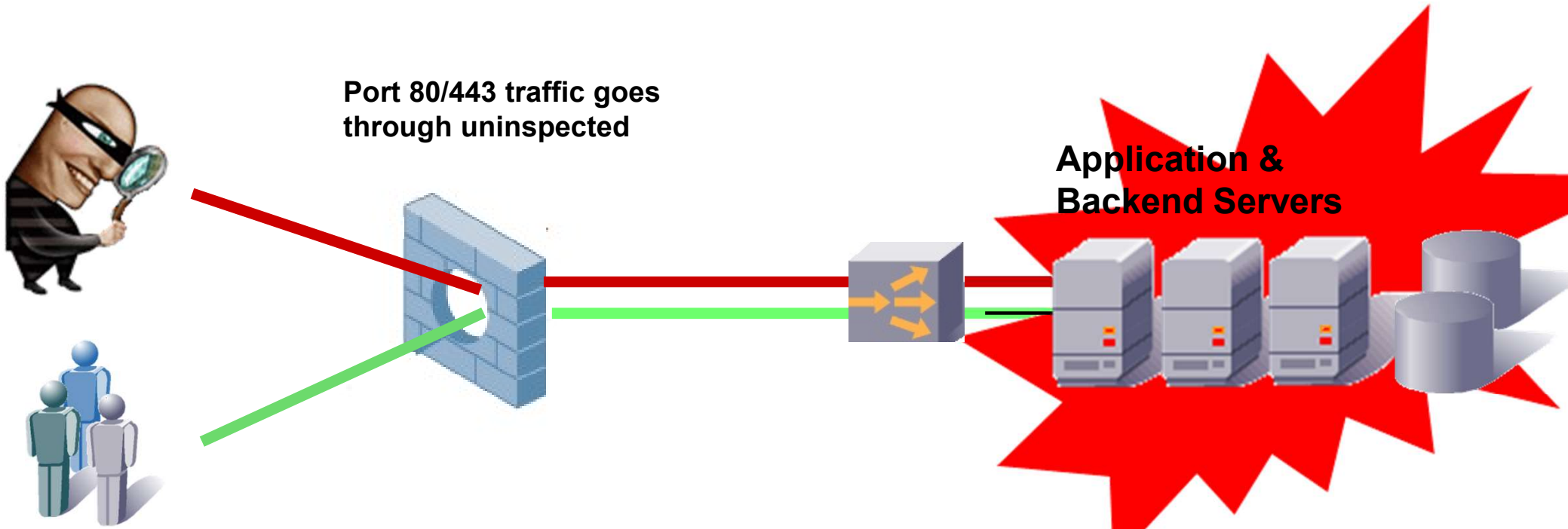
Content

- What is WAF?
- **Why we need to use WAF?**
- Web Application architecture
- How does WAF prevent attacks?
- How to deploy WAF?

Why we need to use WAF?

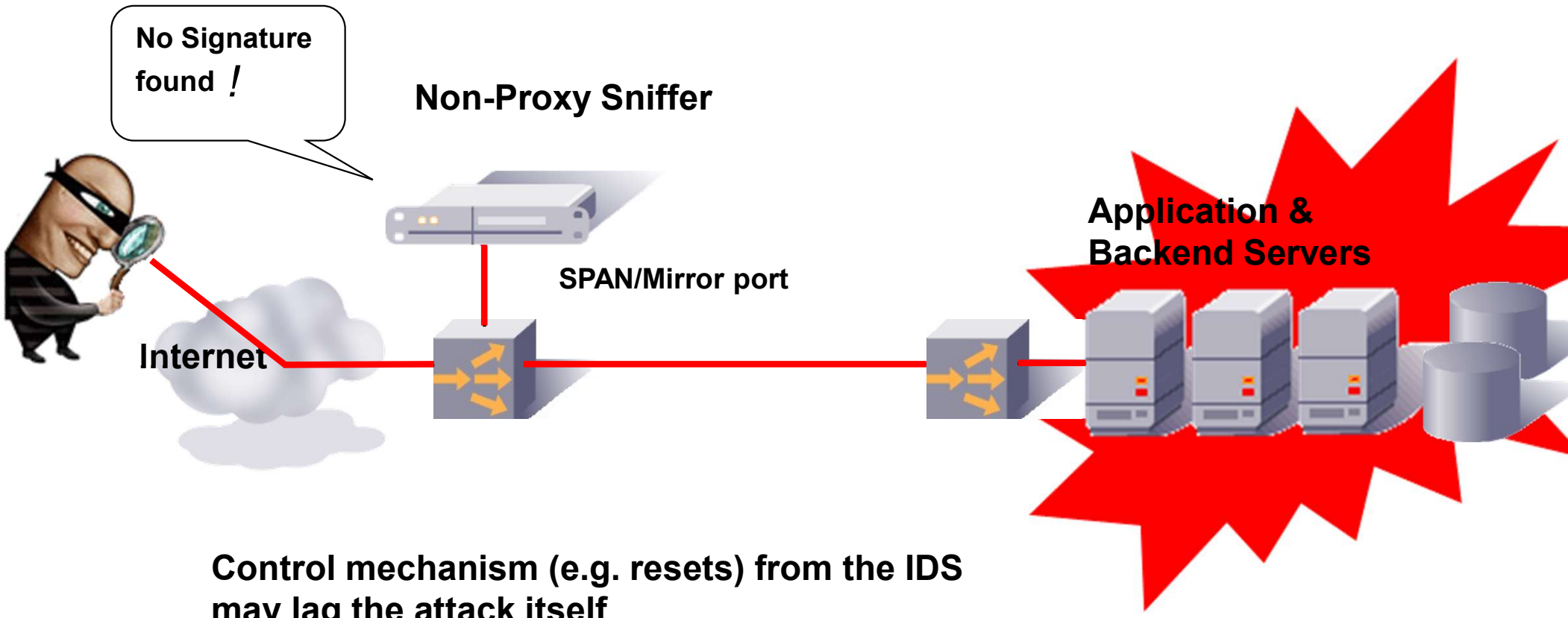


Why Network Firewalls Fail



Even SSL Fails!

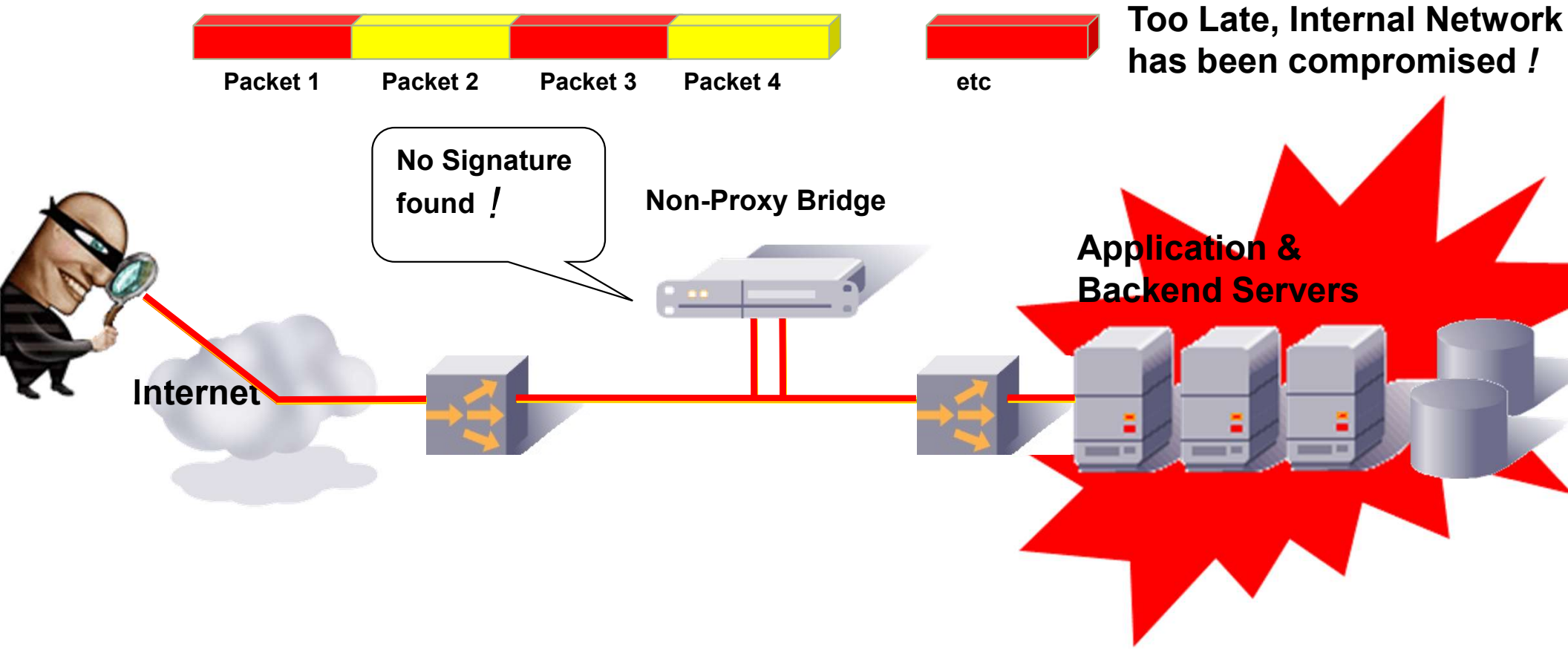
Why Offline IDS/IPS Fail



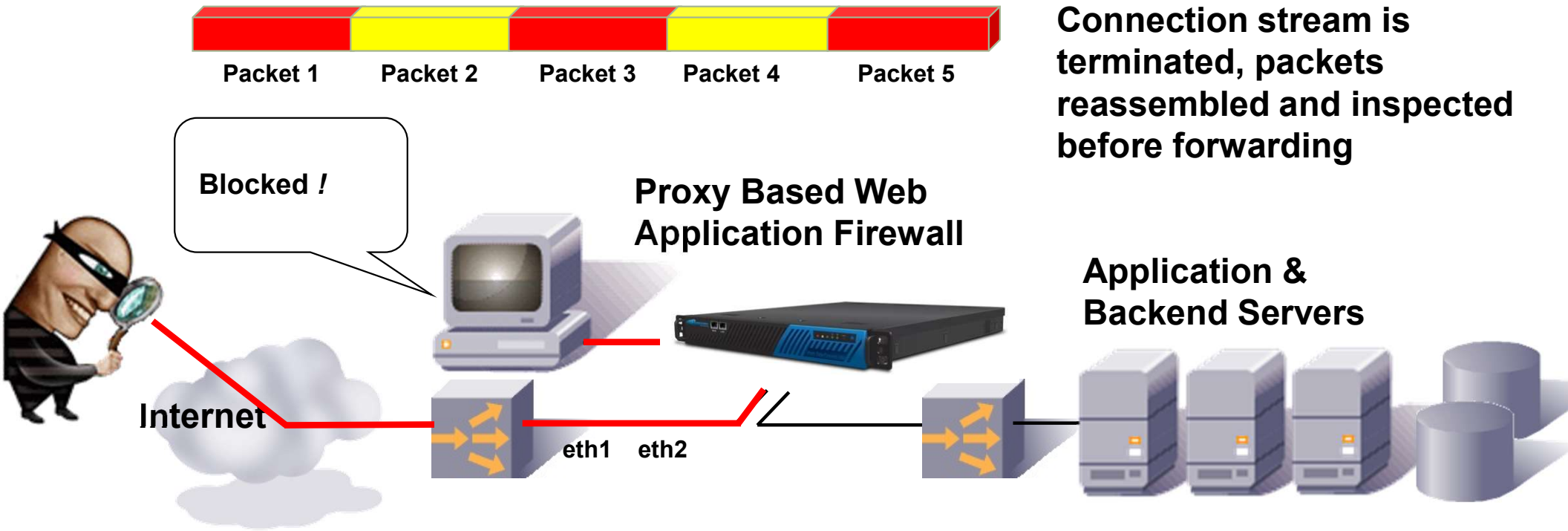
Control mechanism (e.g. resets) from the IDS may lag the attack itself

Signature based security does not protect from *zero day attacks*

Why Non-Proxy Inline Bridge Fails

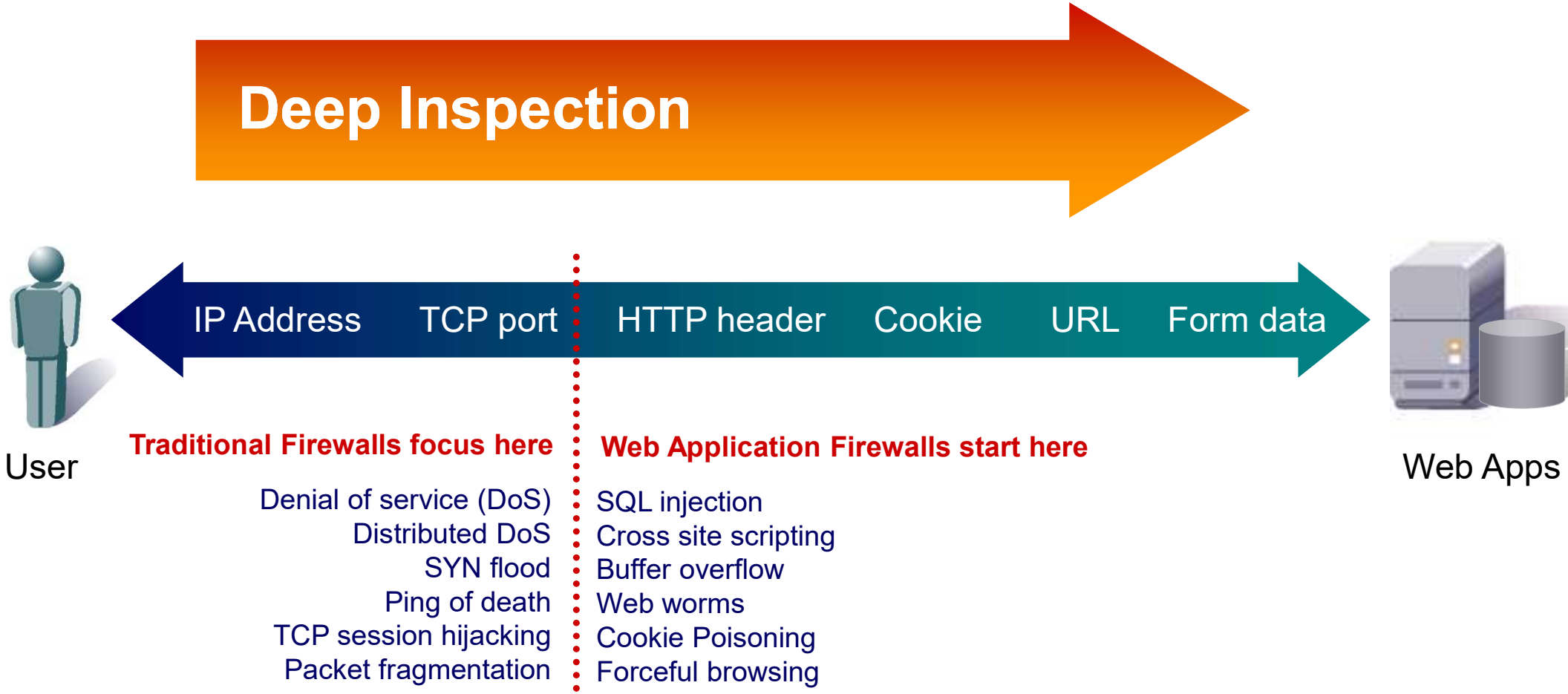


Why Application Firewall Proxies Work



Connection stream is terminated, packets reassembled and inspected before forwarding

WAFs Inspect Application Data



Why Network Firewall/IDS not enough?

Network firewalls and IPS solutions are packet based, not session based - No session state is maintained

Full context of user sessions must be understood to effectively inspect for attacks

IDS monitors network traffic looking for the characteristics of known attacks.

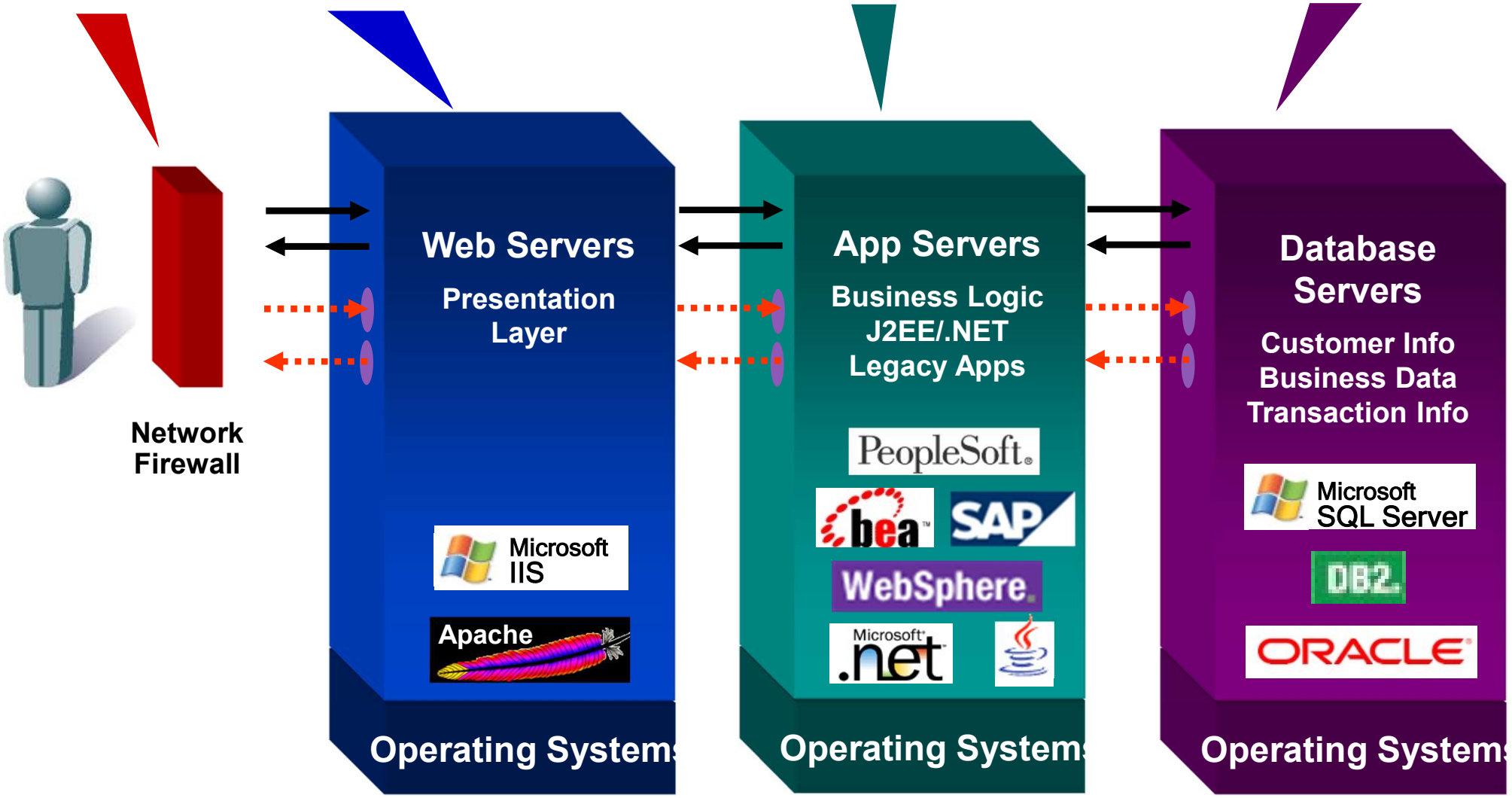
Application Threat	IPS / Network Firewalls	Barracuda Web App Firewall
Cookie poisoning	Well known signatures only	✓
Hidden field manipulation	Well known signatures only	✓
Cross Site scripting	Well known signatures only	✓
SQL and Command Injection	**None**	✓
Stealth Commanding	**None**	✓
Parameter Tampering	**None**	✓
Buffer overflow	**None**	✓
Forceful Browsing	**None**	✓
Identity Theft	**None**	✓
Application DoS	**None**	✓
Data Theft	**None**	✓

Content

- What is WAF?
- Why we need to use WAF?
- **Web Application architecture**
- How does WAF prevent attacks?
- How to deploy WAF?

Web Application architecture

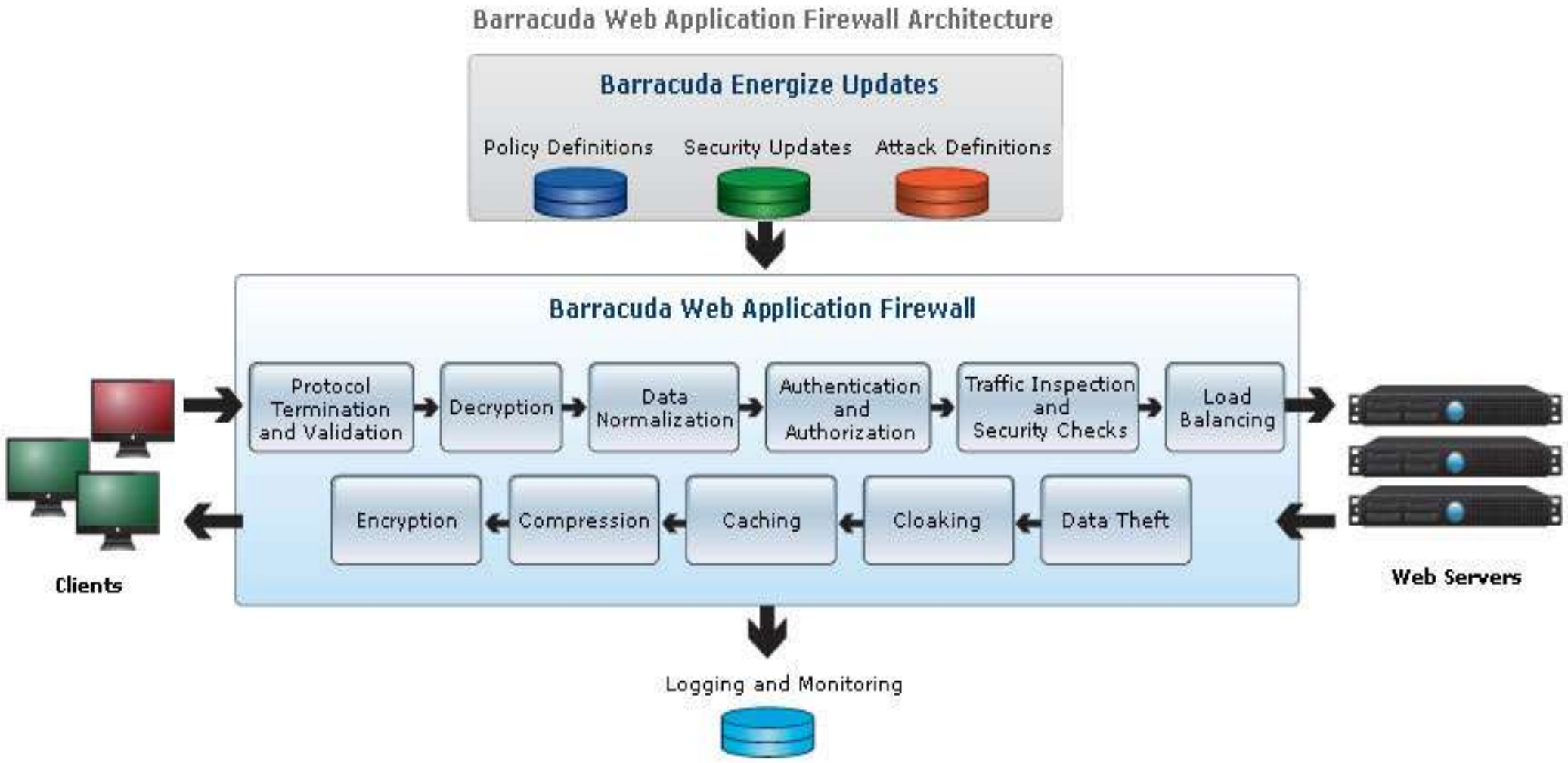
http ://www.none.to/script ?submenu=update&uid =1'+or+like'%25admin%25';--%00

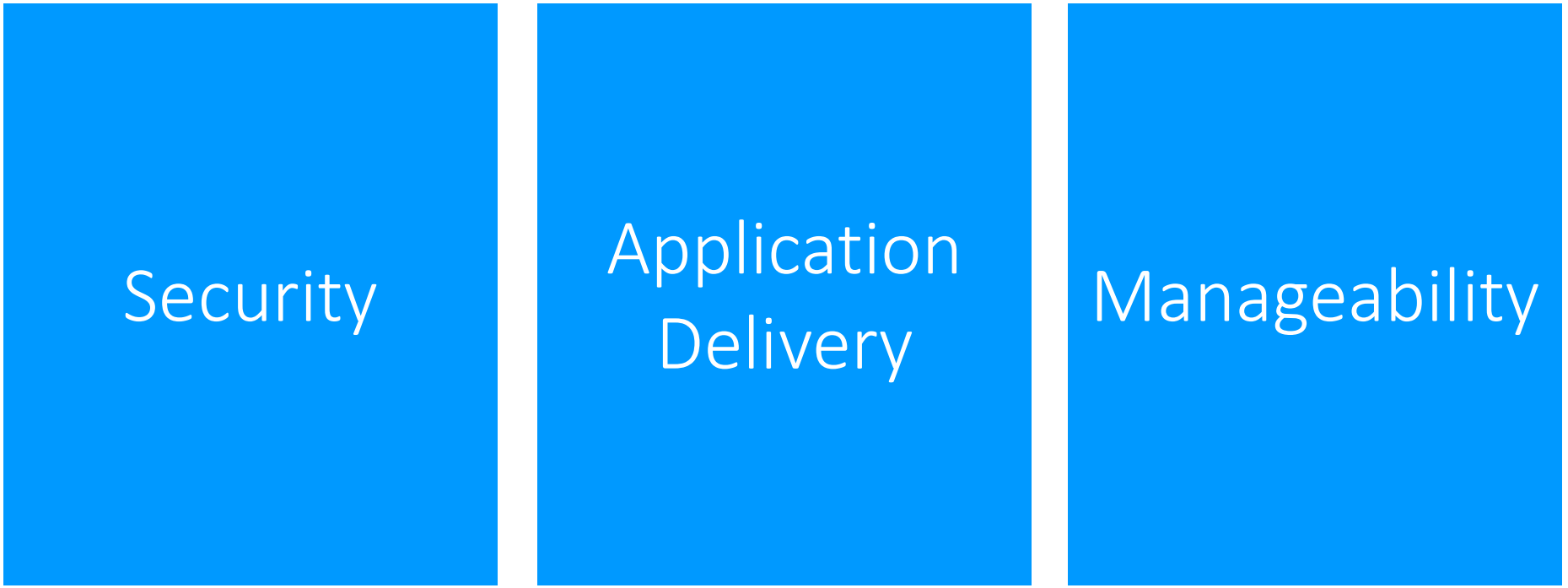


Content

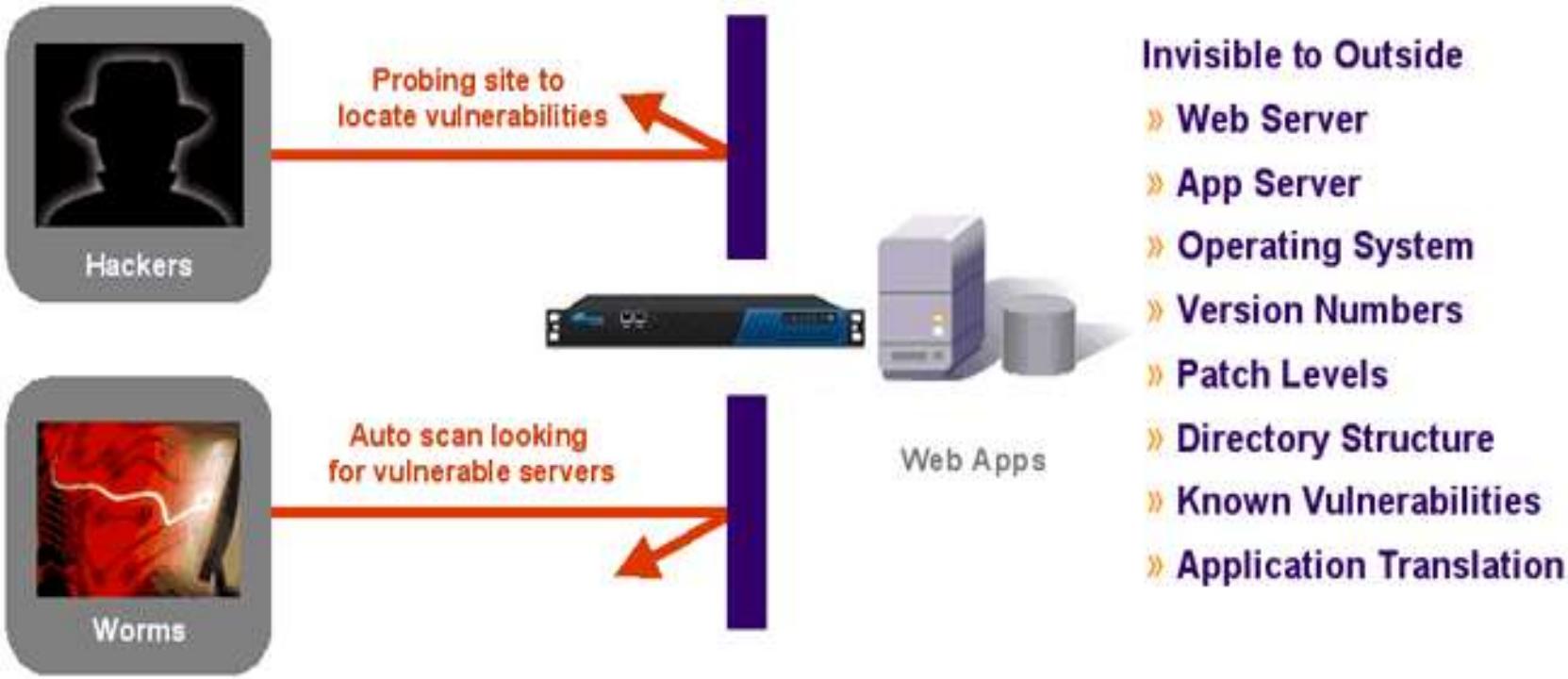
- What is WAF?
- Why we need to use WAF?
- Web Application architecture
- **How does WAF prevent attacks?**
- How to deploy WAF?

How does WAF prevent attacks?




Security
Application Delivery
Manageability

Cloaking



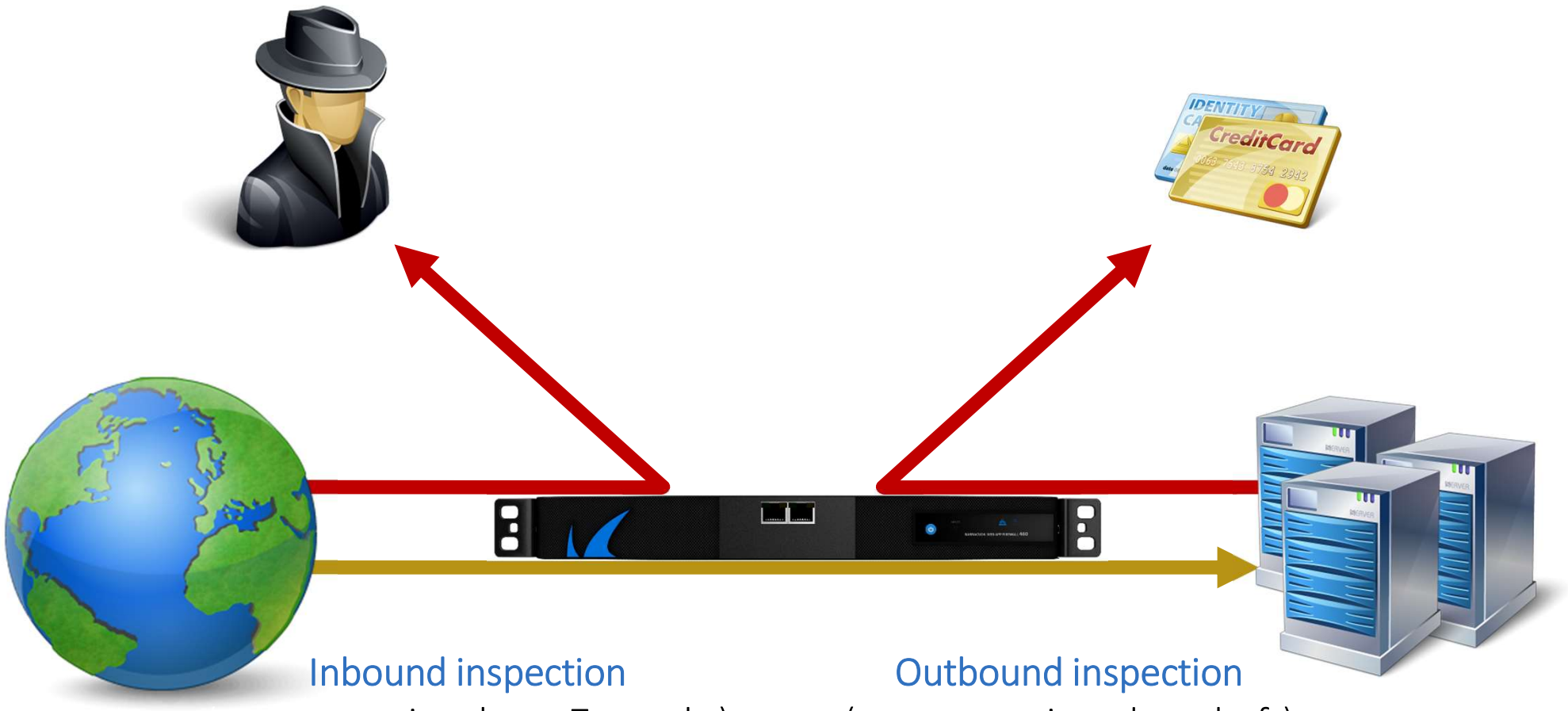
Attackers first task – Reconnaissance of network for weakness

- **What Web, Database, App server etc?**
- **What versions, patches, known vulnerabilities etc?**

Cloaking makes enterprise Web resources invisible to hackers and worms

- **Hides all error codes, HTTP headers, IP addresses etc**

Layer 7 Web Application Firewall



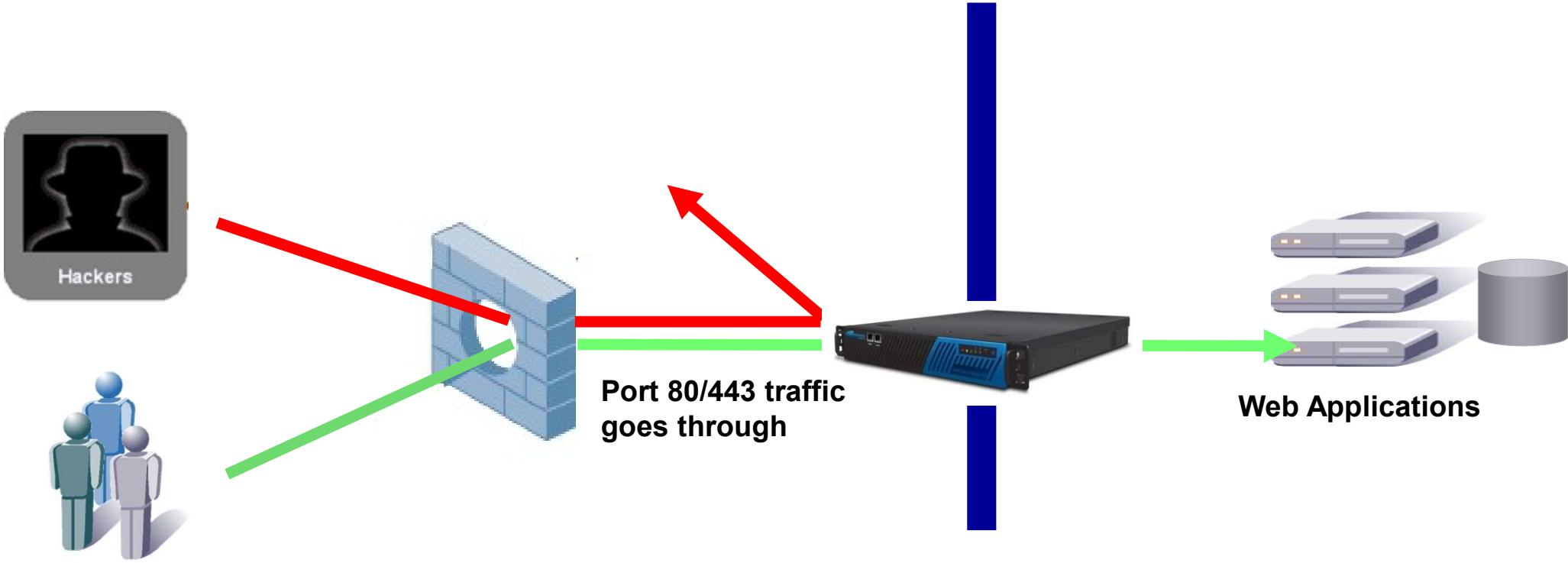
Inbound inspection
protect against layer 7 attacks)

Outbound inspection
(protect against data theft)

Inbound Protection

- Injection – SQL, OS commands etc
- Scripting – XSS, CSRF
- Cookie/Session Poisoning
- Parameter/Form Tampering

- Protocol Sanitization
- Zero day attacks protection
- Anti Virus Attacks prevention
- XML Attacks



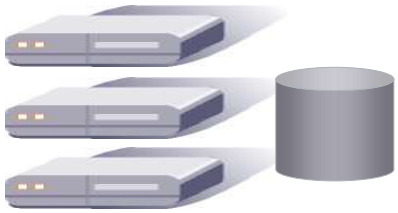
Prevention in TOP 10 OWASP

OWASP Top 10	Prevention
1. Cross Site Scripting (XSS)	Validate inputs for Script injections
2. Injection Flaws	Validate inputs for interpreter injection attacks, particularly SQL injection
3. Malicious File Execution	Prevent remote code execution, root kit installs, and use of file system resources
4. Insecure Direct Object Reference	Learn and enforce only valid direct access to objects, files, directories, database records, URLs and form parameters
5. Cross Site Request Forgery (CSRF)	Insert random character sequence in URLs to prevent against CSRF attacks
6. Info Leakage / Imporper Error Handling	Prevents sensitive data from being exposed
7. Broken Authentication and Session Management	Enforce proper authentication, protection of session tokens.
8. Insecure Cryptographic Storage	Securely store the private /confidential data
9. Insecure Communications	Enable encryption for the communication channels
10. Failure to restrict URL Access	Enforce URL access by Learning and allowing access to valid URLs. This can be further tuned by configuring authorization rules via the AAA functionality

Outbound Protection

Deep Inspection of outgoing content blocks:

- **Credit Cards**
- **Social Security Numbers**
- **Custom Patterns**
- **Error details in 200 OK responses**

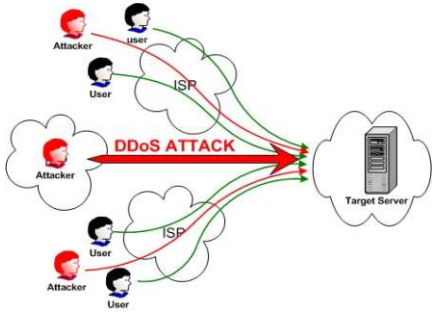


Web Applications

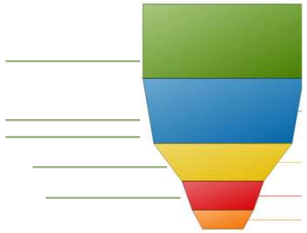
Advanced Application Security

Anti Virus Protection

XML Firewall

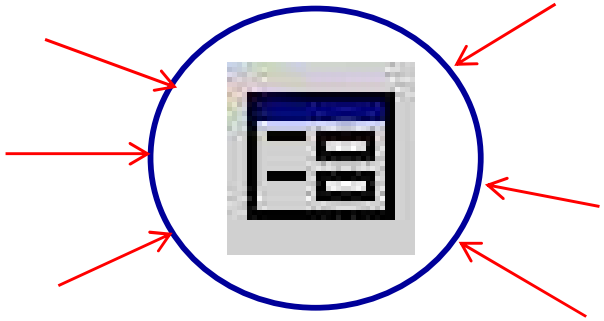


Session Tracking



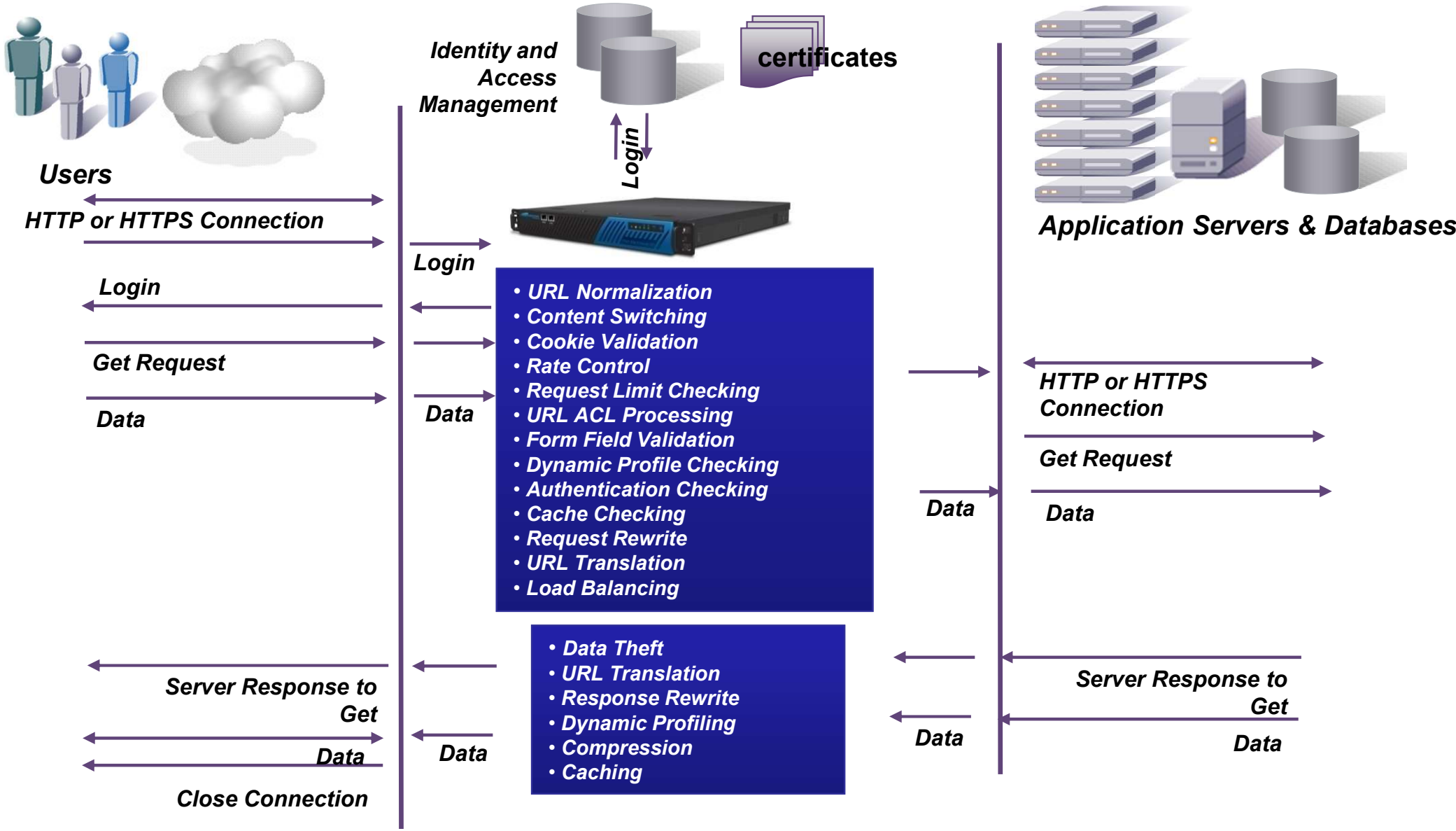
Rate Control

Adaptive Profiling



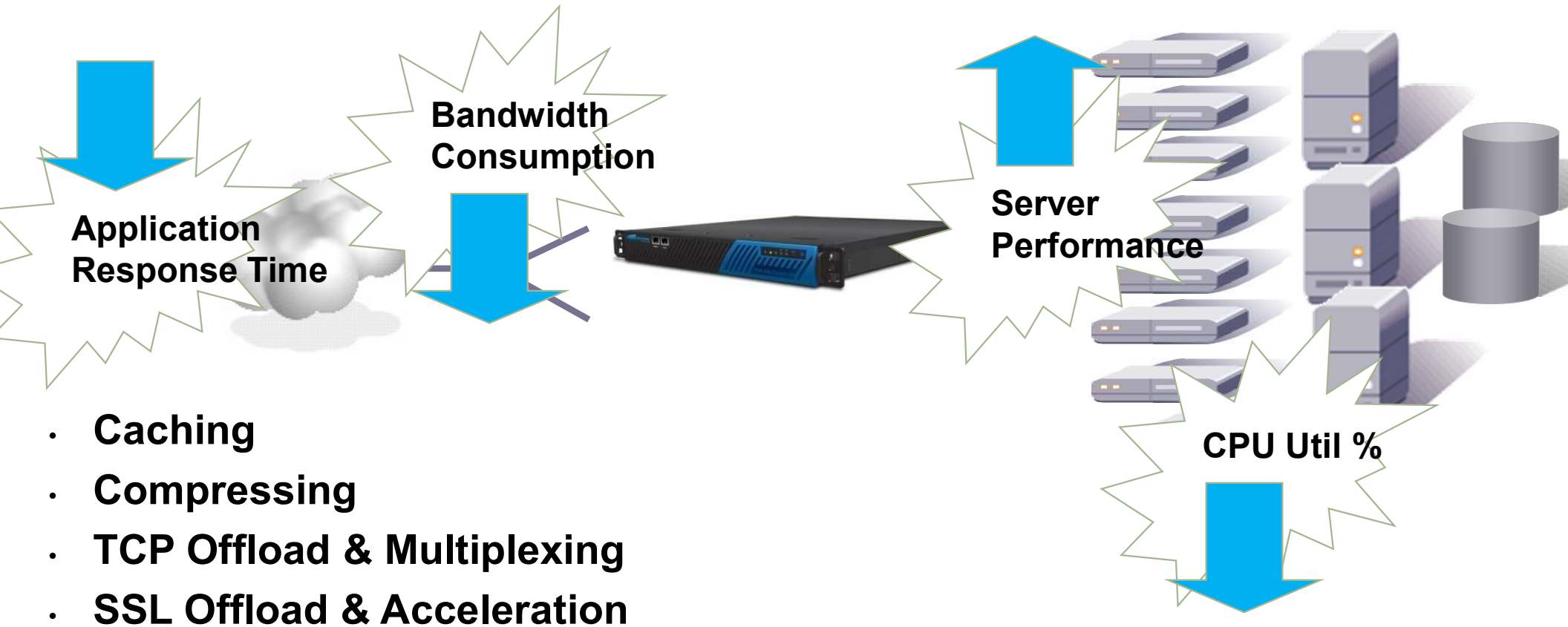
Bruteforce prevention

Bi-Directional Detection and Mitigation




Application Delivery

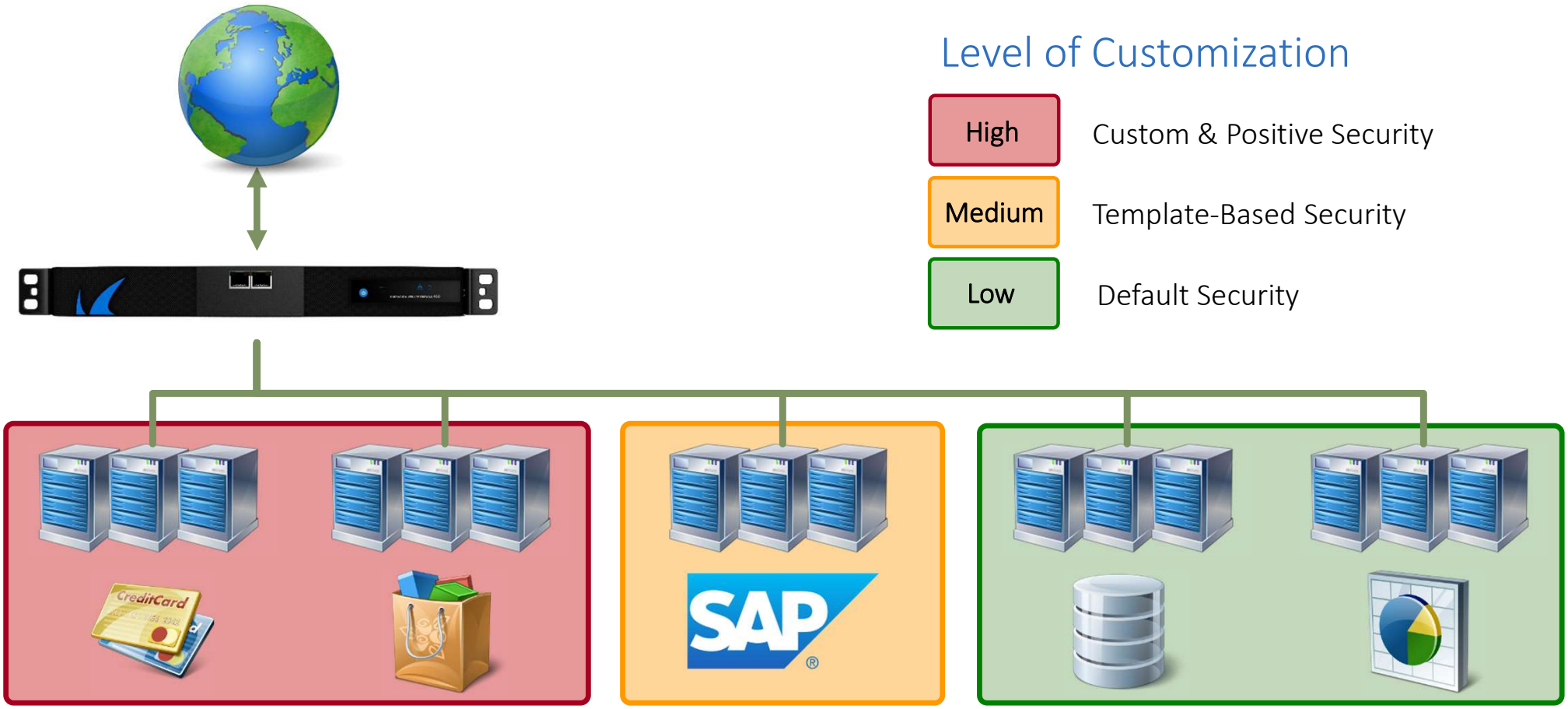
Also Improves Operational Efficiency



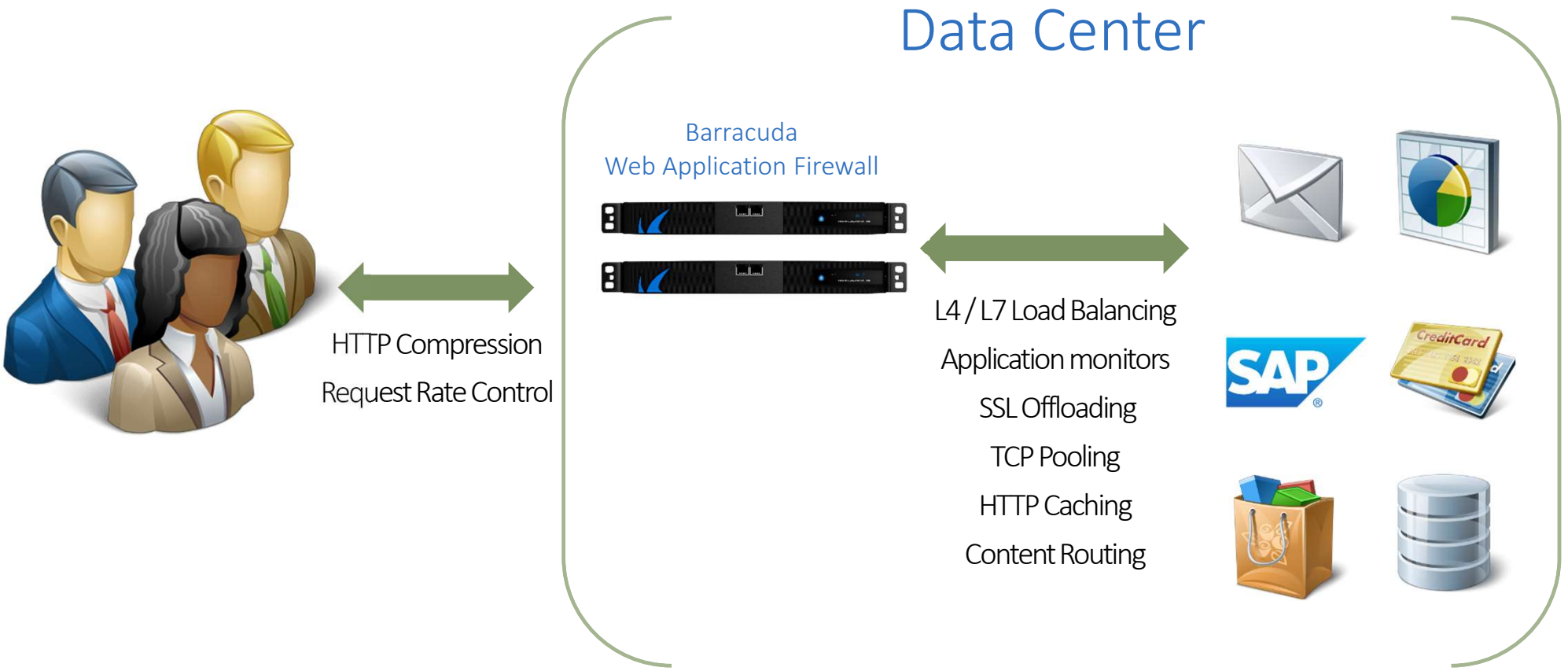
- Caching
- Compressing
- TCP Offload & Multiplexing
- SSL Offload & Acceleration
- Load Balancing

30 – 400% Response Time Improvement Plus Complete Application Security

Plug & Play Deployment & Management

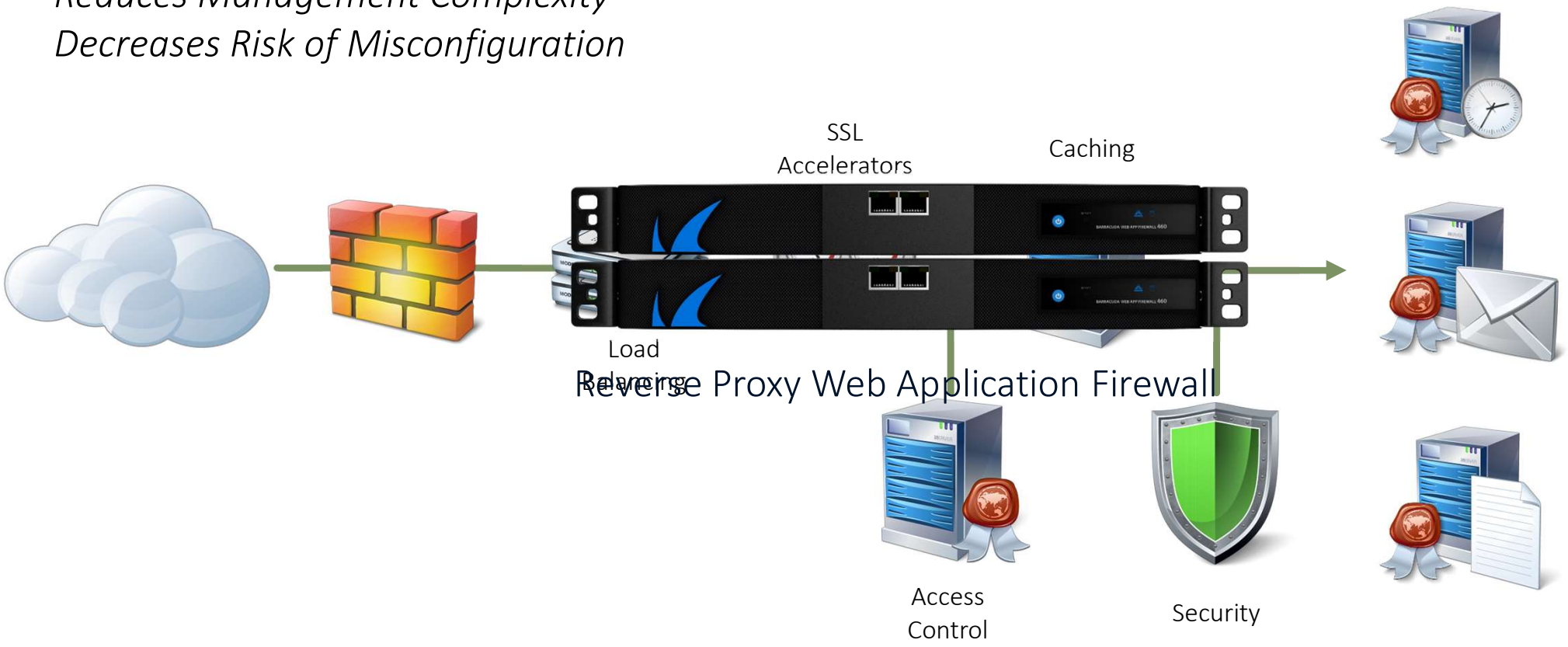


Acceleration & Load Balancing



Consolidate Disparate Appliances in DMZ

Reduces Management Complexity
Decreases Risk of Misconfiguration



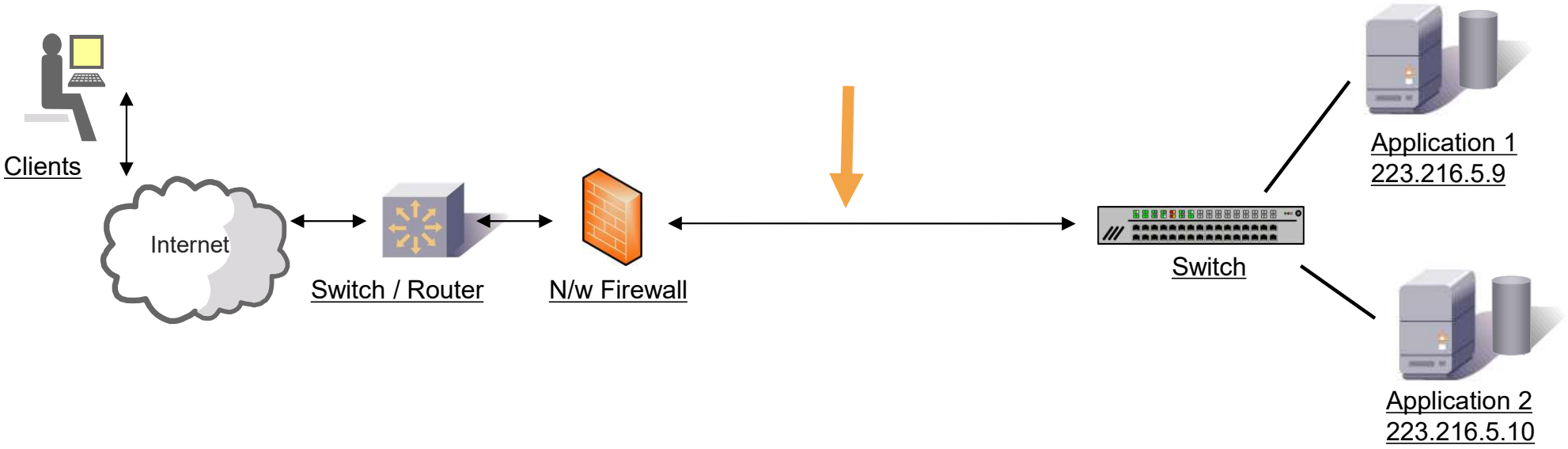
Content

- What is WAF?
- Why we need to use WAF?
- Web Application architecture
- How does WAF prevent attacks?
- **How to deploy WAF?**

Bridge Mode

Existing Network/Application Data Flow

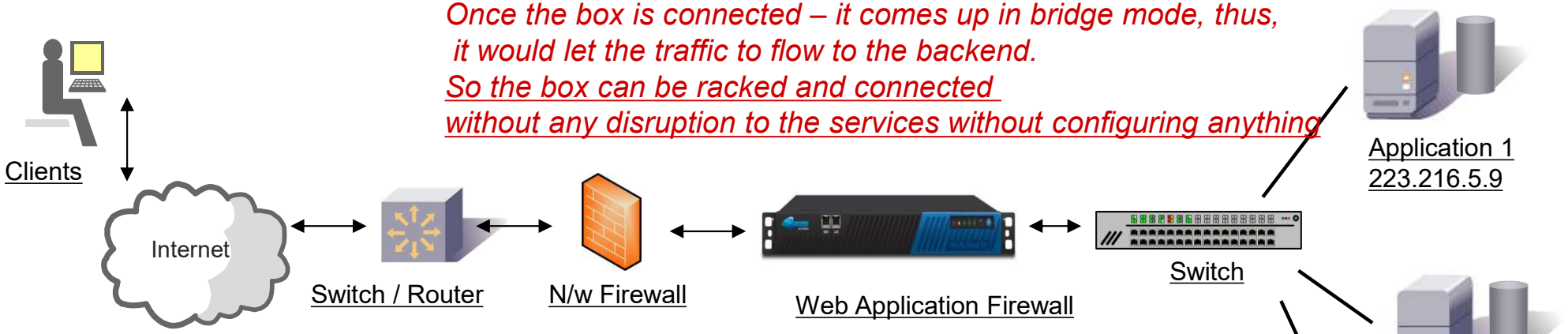
The Barracuda Web Application Firewall is inserted between the Network firewall and the switch to the backend.



Connecting in Bridge Mode

The Barracuda Web Application Firewall is inserted between the Network firewall and the switch to the backend.

Once the box is connected – it comes up in bridge mode, thus, it would let the traffic to flow to the backend.
So the box can be racked and connected without any disruption to the services without configuring anything



The IP address provided during the initial set up screen is the one used to access the management Interface.

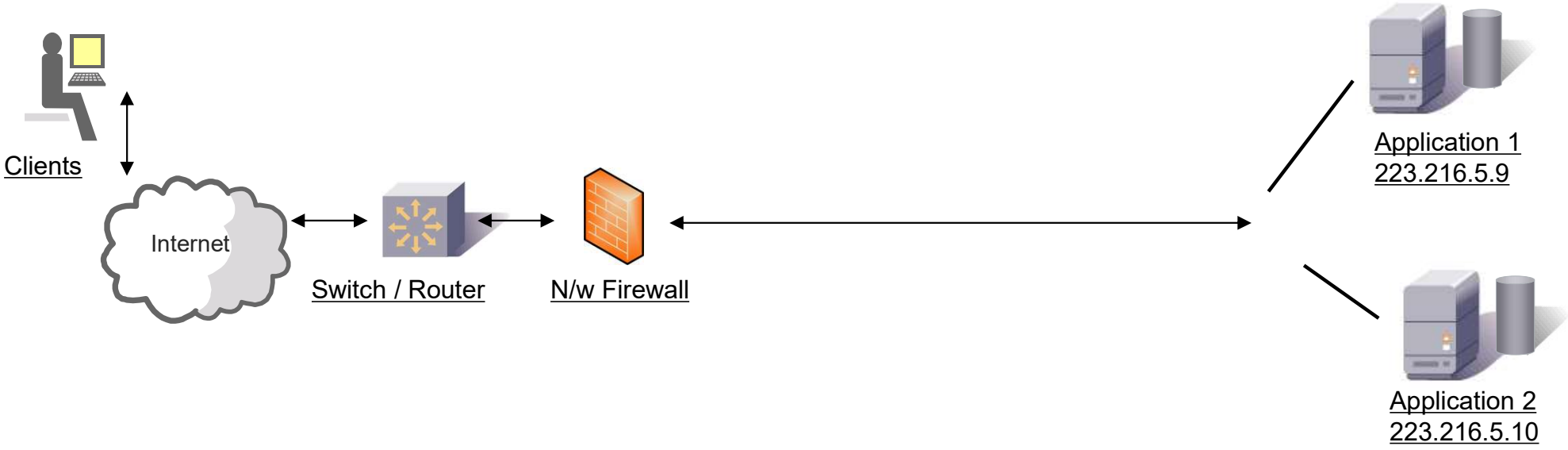
The IP address of the box should be reachable from the place you are going to do the configuration from.

The web GUI is available at *http://<IP Address>:8000*

Proxy mode

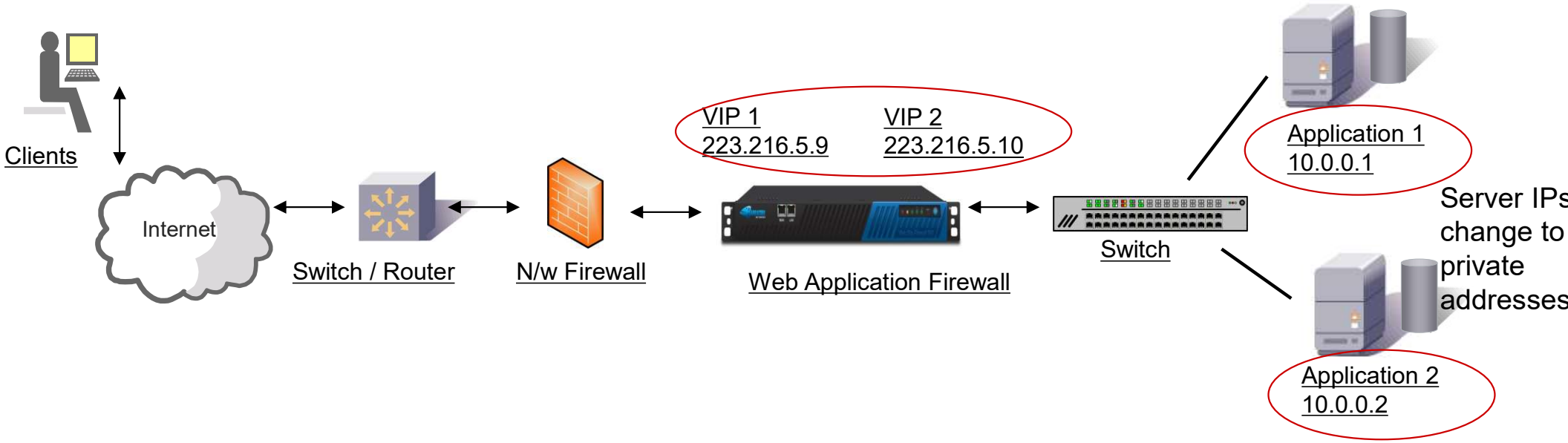
Existing Network/Application Data Flow

The Barracuda Web Application Firewall is inserted between the Network firewall and the switch to the backend.



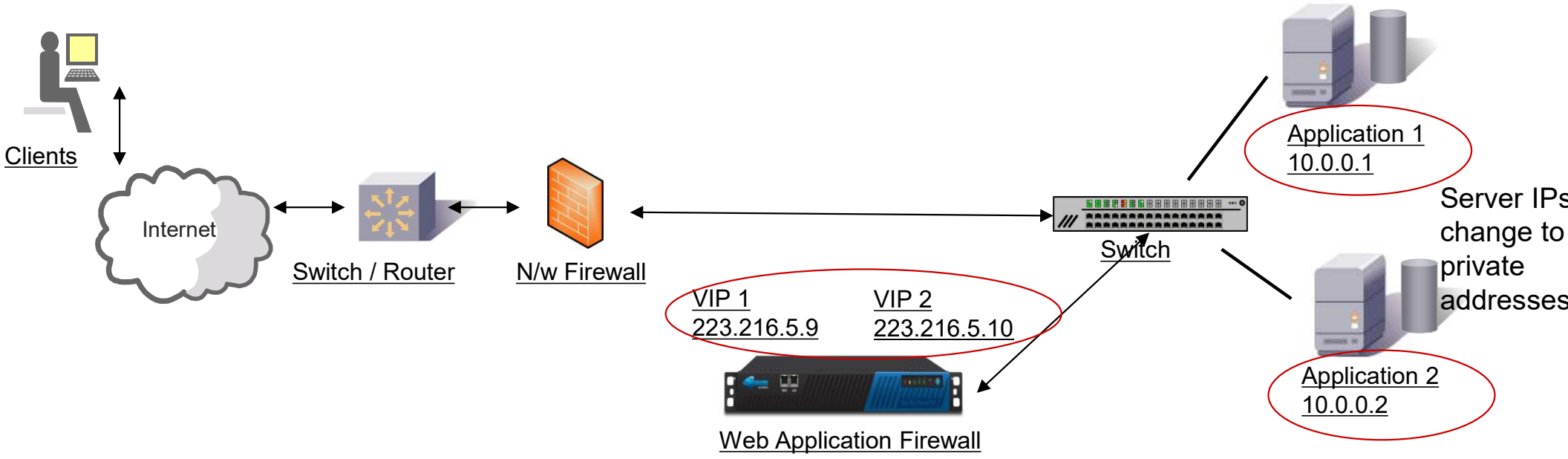
Connecting in Proxy Mode

The Barracuda Web Application Firewall is inserted between the Network firewall and the switch to the backend.



One-Arm Proxy mode

The Barracuda Web Application Firewall is inserted between the Network firewall and the switch to the backend.



Question ???