

Zabezpečení podnikových aplikací

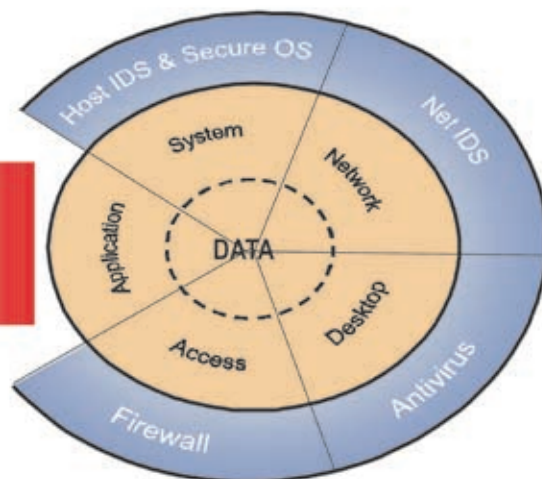
před útoky z internetu

Jsme svědky masivního vystavování podnikových aplikací do prostředí internetu a z druhé strany narůstajícího počtu skrytých hrozeb, proti kterým je stále obtížnější bojovat.

Security's Gaping Hole

"64% of the 10 million security incidents tracked targeted port 80."

Information Week



V praxi se velmi často můžeme setkat s případem, kdy klient řeší dilema typu – mám aplikaci, která je podpůrným nástrojem pro komunikaci s okolním světem a tudíž ji nemůžu jen tak vypnout. Mám výsledky nedávno provedených bezpečnostních testů, odkrývající nedostatky v jejím zabezpečení a odvozené možné následky, plynoucí z nepokrytí jednotlivých rizik.

Smluvně definované požadavky na bezpečnost zákazníků nemá a dodavatel aplikace se netváří, že by jen tak „zdarma“ chtěl přepsat kód v aplikaci. Horším případem může být, že aplikace je tak zastaralá, že původní dodavatel již neexistuje nebo to není v jeho současných silách.

Zákazník tak stojí před těžkým úkolem, jak co nejrychleji a s přijatelnými náklady řídit rizika související s provozem svých podnikových aplikací v prostředí webu.

Nedostatky současné situace

V současné době se často potýkáme s odpověďmi typu: je to za firewallem, máme IDP/IPS, používáme SSL, běží to na bezpečném OS, aplikovali jsme patche, uživatel musí mít

heslo. Nic z toho, ale aplikační bezpečnost primárně neřeší. Tyto jednotlivé části podporují celkovou úroveň bezpečnosti, ale nechrání samotnou aplikaci.

- Šifrování SSL není dostatečné – zabezpečuje komunikaci, ale na serveru musí být dešifrovány. Navíc není zpravidla používána autentizace uživatele certifikátem, takže se serverem může šifrovaným kanálem komunikovat kdokoli.
- Samotný firewall nestačí – 1) musí být otevřeny porty pro přístup na WWW servery (80, 443). 2) Musí být otevřený pro komunikaci zevnitř – útoky Trojského koně, kdy program útočí z Vašich systémů firewall nezachytí.
- Standardní skenovací nástroje nestačí – sledují známé zranitelnosti, otevřené porty atd., ale nezjišťují chyby v kódu na Web a aplikačních serverech.
- IDS/IPS zpravidla sledují pouze útoky na známé zranitelnosti, skenování portů apod. Pokud útočník přistupuje jako uživatel přímo k aplikaci, většinou jej IDS/IPS nezaznamená (výjimkou může být velký počet pokusů o spuštění kódu apod.)

Bezpečnost aplikační vrstvy se dnes zpravidla řeší značně intuitivně nebo v horším případě vůbec. Reálným dopadem jsou poměrně vážné bezpečnostní chyby v provozovaných aplikacích.

Rychlou a účinnou eliminací rizik je Webový aplikační firewall/XML firewall od společnosti F5. Nabízí kompletní ochranu proti nejběžnějším a nejnebezpečnějším hrozbám souvisejícím s webovými aplikacemi i aplikacemi webových služeb používajících jazyk XML a úplnou ochranu proti odcizení dat.

Rychlá eliminace chyb dle projektu OWASP-Open Web Application Security Project Poskytuje schopnosti učení, díky kterým je poskytována ochrana proti sofistikovanějším útokům na aplikace, nabízí ochranu proti útokům zaměřeným na konkrétní uživatelské relace (nutné pro aplikace e-commerce, zabezpečený extranet a online bankovníctví) a poskytuje přesnější kontrolu zásad zabezpečení aplikací.

Řešení od F5 je k dispozici na více hardwarových platformách, aby bylo možné splnit požadavky na výkon a dostupnost jakékoli

organizace – od malého podniku po velká datová střediska.

Lze jej pořídit jako samostatnou appliance ASM 3600, 6900, 8900 nebo jako rozšiřující SW modul k platformě BIG-IP LTM či

bezpečnostní patch nebo nasazení nové verze aplikace. Jedná se o případ, kdy je publikován nový útok a kdy výrobce nebo dodavatel není schopen v krátké časové době připravit a distribuovat opravu.

nápravu bez důkladnějšího studia chyby. Příkladem může být chyba v administracním rozhraní, které je určené jak pro externí, tak interní uživatele. Dočasným řešením může být povolení této funkce pouze pro interní administrátory, kteří jsou považováni za důvěryhodné, při zachování zbylé funkcionality.

• „Hardening“ webových aplikací

V neposlední řadě může být WAF poslední záchrannou brzdou, pakliže dodavatel, respektive vývojář, prakticky netuší, co znamenají zkratky a termíny jako XSS (cross site scripting) či SQL injection. Validaci dat a jejich jednotné kódování považuje za zbytečné a bezpečnost chápe jako použití SSL protokolu. V tomto případě opravdu není reálné počítat s tím, že by něco dokázal opravit a náhrada aplikace ze dne na den není většinou možná.

Více informací naleznete na www.f5.com

VIPRION, díky níž je možné z jednoho bodu řešit otázku distribuce zátěže mezi webovou farmou, akcelarovat výkon webových aplikací a zvýšit bezpečnost aplikací.

Řešení od F5 splňují požadavky Common Criteria Certification a PCI DSS 1.2.

Klíčové vlastnosti

• Detekce anomálií v HTTP provozu, následná ochrana a logování

Webový aplikační firewall (dále jen WAF) je schopen analyzovat na aplikační vrstvě veškeré náležitosti HTTP protokolu.

Je použitelný také pro ochranu tzv. web services. Může analyzovat jak záhlaví, tak tělo dotazů, a to samé platí také pro odpovědi. Proti nejčastějším útokům lze s úspěchem používat dodávaná pravidla, tzv. core rules, pro složitější pravidla je k dispozici jazyk pro definici pravidel. Po vyhodnocení daného pravidla může reagovat jako standardní firewall – od zahazení požadavku přes zobrazení chybového hlášení, přesměrování na tzv. honeypot až po prosté zalogování události.

• Just-in-time patching

Existují dva základní důvody, proč je nutné, nebo přinejmenším vhodné mít k dispozici rychlé řešení, kterým lze nahradit například

Druhým důvodem je objevení bezpečnostní chyby v rámci vlastní provozované aplikace, kdy může být problém realizovat rychlou

Tradiční bezpečnostní nástroje vs. Webový aplikační firewall F5

	Network Firewall	IPS	ASm
Know Web Worms	Limited	✓	✓
Unknown Web Worms	✗	Limited	✓
Known Web Vulnerabilities	Limited	Partial	✓
Unknown Web Vulnerabilities	✗	Limited	✓
Illegal Access Web	Limited	✗	✓
Forceful Browsing Vulnerabilities	✗	✗	✓
File/Directory Enumerations	✗	Limited	✓
Buffer Overflow	Limited	Limited	✓
Cross-Site Scripting	Limited	Limited	✓
SQL/OS Injection	✗	Limited	✓
Cookie Poisoning	✗	✗	✓
Hidden-Field Manipulation	✗	✗	✓
Parameter Tampering	✗	✗	✓
Layer 7 DoS Attacks	✗	✗	✓
Bruteforce Logging Attacks	✗	✗	✓
App. Security and Acceleration	✗	✗	