理机操作系统由 Redhat,Centos,Windows Server 2008 等操作系统构成可选集.另一种是人为异构化,对于 sql 查询语句、文件(包括文件目录)等不具备多样性的数据,采用了关键字标签化、文件标签化、目录随机化等方法,使执行体上的数据具备人为的异构性.基于多样性产生的异构性在一定程度上存在“同源”的问题,相对而言,人为异构化产生的异构性更强.人为异构化的方法多种多样,根据不同的对象形式和特点,通过较小的改变可以取得较大的异构性,异构化方法相对私密,破解难度较大,因而,人为异构化方法能够通过较小的代价实现较高的异构性,从而增强拟态防御模型的内生安全性.

动态控制模块对系统整体进行管理控制,包含了动态选择这一拟态防御模型的关键机制,主要功能包括改变比较表决策略、更改系统或软件的安全性配置、替换本层失效构件、控制执行体集的轮换、清洗下线执行体使其恢复到安全状态等.动态选择机制接收各层的异常反馈信息,如构件运行状态.同时,主要依据响应裁决模块的不一致信息反馈判断异常或失效的执行体,适应性地对异常的执行体进行替换,或依据系统受损程度对执行体集进行替换.除此之外,在无异常情况下进行执行体集的随机变化,增大系统的不确定性,增加攻击难度,在一定程度上抵御长期持续的探测或攻击,如 APT 攻击.

拟态 Web 服务器的原型实现中有两个表决器:一个是在响应返回给用户之前设置表决器进行表决,得到唯一响应,记为前端表决器.另一个是在服务器与数据库之间,对服务器输入数据库的 sql 查询语句进行一次表决,以过滤非法 sql 语句,记为后端表决器.前端表决是在语义层面进行表决,由于不同类型服务器的响应在数据层面上有细节的差异,如包头的 context-type 域、server 域的差异,这些差异属于非语义的差异;又如内容中包含的空格、“/”等符号的个数的差别.为了避免这些差异影响表决结果,前端表决在语义层面进行.后端表决的对象是从多个冗余服务器输入数据库的 sql 语句,sql 语句的异构化由拟态异构化机制实现,是一种不改变语义的数据层异构化,因而在该层面的表决是数据层次的表决.

5 实验评估

为了检验拟态防御 Web 服务器的有效性,对其进行了多方面的测试.以安全性测试为主,设置参照对象,进行模拟攻击对比测试.在此基础上,对拟态防御 Web 服务器进行性能测试,评估其性能损耗.

5.1 安全性测试

安全性测试主要从远程攻击和内网渗透两方面进行,模拟攻击覆盖了入侵链的大部分环节,对比了拟态防御 Web 服务器(device under test-1,简称 DUT-1)、具备典型安全防护的服务器(DUT-2)和无安全防护服务器(DUT-3)在同样攻击下的防御效果.

3 种实验对象,除了拟态防御 Web 服务器为实现多样性和异构性而同时具有多种虚拟操作系统外,其余主机操作系统、服务器软件、Web 应用脚本均一致,典型安全防护 Web 服务器在无安全防护服务器的基础上增加了卡巴斯基安全软件、安全狗等安全工具.3 种实验对象的组成配置见表 2.

Table 2 Configuration of the devices under test

表 2 测试对象配置

实验对象	主机操作系统	虚拟操作系统	服务器软件	脚本、数据	其他安全工具
DUT-1	Windows server 2008 R2, Centos 7	Windows server 2008 R2, Ubuntu 14.04, Centos 6.6, Windows server 2003, Windows XP SP3, Windows 7	Apache, Nginx, IIS, Lighttpd	异构化处理后的脚本和数据	无
DUT-2	Windows server 2008 R2	Windows XP	Apache2.0.63	未异构化处理的脚本和数据	卡巴斯基, 安全狗
DUT-3	Windows server 2008 R2	Windows XP	Apache2.0.63	未异构化处理的脚本和数据	无

在 3 种测试对象上分别安装相同的测试网站,安全配置完全一致.设计的测试案例覆盖了攻击链的各个阶段,共包含 15 项测试.

在扫描探测阶段,使用 Nikto,Nmap 等扫描工具,并利用绿盟远程安全评估系统进行多次扫描和探测,统计扫

描结果,验证拟态防御技术对系统信息呈现性质的影响.

在漏洞挖掘阶段,在已知系统内部含有 Apache 目录浏览漏洞的前提下,尝试利用该漏洞获取网站目录.另外,通过 Burpsuite 等工具截获并修改数据包,诱使访问出错,尝试通过显示的错误路径获取系统后台信息.通过以上测试评估拟态防御技术对漏洞挖掘阶段的防御效果.

在攻击植入阶段,通过利用系统局部的已知漏洞或故意制造漏洞,模拟漏洞利用、木马植入、病毒感染、sql 注入等攻击植入过程.需要特别说明的是,在测试中,虚拟机的 Windows 操作系统保留了 ms08_067 和 ms12_020 两个漏洞,模拟对防御者而言的未知漏洞,并通过预埋后门模拟对防御者而言未知的后门.通过以上测试,评估拟态防御技术对利用已知和未知漏洞后门的攻击的防御能力.另外,通过系统局部宕机评估拟态防御技术对除攻击以外的异常情况的应对能力.

在攻击维持阶段,预埋后门,并修改攻击的表决结果,故意使第 1 次利用后门的攻击成功,而后再次利用后门,检验攻击结果.通过这种测试,验证拟态防御技术对攻击维持阶段的防御效果.

以上测试案例的设计多采用人为预置漏洞或后门的方法,以保证攻击的直接性和破坏性.测试类型遍布攻击的各个阶段,覆盖不同类型的攻击方式,从而在最大范围内、最大程度地对拟态防御 Web 服务器的防御能力进行测试.测试案例及结果见表 3.

Table 3 Testing results of security

表 3 安全性测试结果

攻击阶段	测试项	DUT-1	DUT-2	DUT-3
扫描探测	1. Nikto,Nmap 扫描 2. 绿盟远程安全评估系统扫描	多次扫描探测 得到的信息不一致	多次扫描探测 得到的信息一致	多次扫描探测 得到的信息一致
漏洞挖掘	3. 尝试浏览 Apache 目录	失败	成功	成功
	4. 诱导访问出错,暴露路径	失败	成功	成功
攻击植入	5. 篡改网页	失败	成功	成功
	6. 触发“菜刀一句话”木马	失败	失败	成功
	7. 触发 Weevely PHP 木马	失败	失败	成功
	8. 触发预埋后门	失败	成功	成功
	9. 利用 ms12_020 漏洞	失败	成功	成功
	10. 利用 ms08_067 漏洞	失败	成功	成功
	11. Win32.Alcaul 等病毒感染	失败	失败	成功
	12. sql 注入	失败	成功	成功
	13. 物理主机宕机	失败	成功	成功
	14. Apache DoS	失败	成功	成功
攻击维持	15. 预埋能够攻击成功的后门,并再次触发	失败	成功	成功
总计	15 项	0 项成功	10 项成功	13 项成功

根据表 3 的测试结果可以发现,拟态防御 Web 服务器可以全面实现对攻击的防御目的.具体而言:在扫描探测阶段,拟态防御 Web 服务器(DUT-1)能够变换系统指纹信息,呈现不确定性;在漏洞挖掘阶段,使漏洞的出现具有不确定性,增大了漏洞利用的难度;在攻击植入阶段,无论是针对已知漏洞的攻击还是针对未知漏洞后门的攻击,均有强抵抗力.同时,在服务可靠性方面也强于 DUT-2 和 DUT-3;在攻击维持阶段,拟态防御 Web 服务器能够持续抵御利用未知后门实施的攻击.

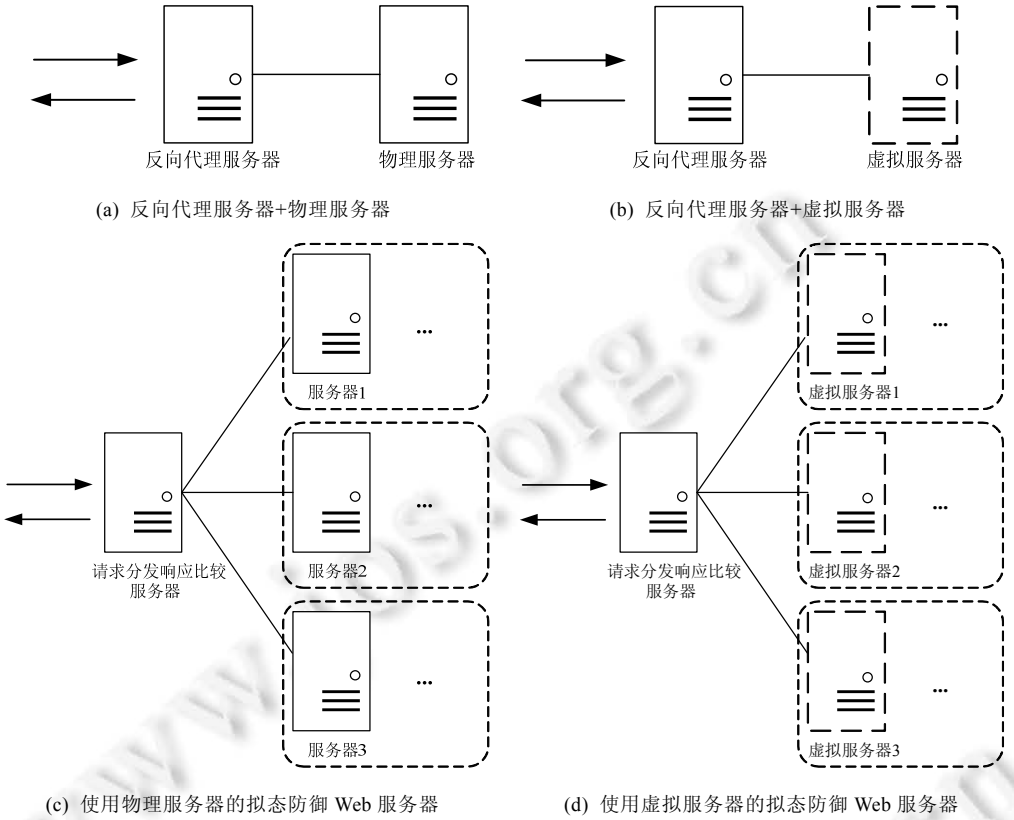
综合以上结果可以得出结论:相比典型的安全防护 Web 服务器和无防护 Web 服务器,拟态防御 Web 服务器在所有攻击案例中均防御成功,安全性最强.

5.2 性能测试

在安全性测试的基础上,测试拟态防御 Web 服务器的性能.将新建速率、并发数、吞吐量和响应时间这 4 项主要指标作为性能评价的考察因素.为了更好地说明性能损耗的原因,采用了对照测试的方法分析拟态防御 Web 服务器的性能代价.如图 8 所示,构建了包括拟态防御 Web 服务器在内的 4 种测试对象.

图 8(a)使用一台反向代理服务器连接物理服务器代表典型的服务器结构,作为基本参照;图 8(b)使用虚拟

机承载的服务器,测试虚拟机服务器的性能;图 8(c)使用物理服务器实现拟态防御 Web 服务器的功能;图 8(d)为拟态防御 Web 服务器的数据流示意图,省略了拟态防御 Web 服务器的感知变迁器等细节,主要用于介绍性能测试的结构.4 种测试对象除了结构不同以外,其余的主机型号、操作系统版本、服务器软件版本等配置对应相同.



(a) 反向代理服务器+物理服务器 (b) 反向代理服务器+虚拟服务器

(c) 使用物理服务器的拟态防御 Web 服务器 (d) 使用虚拟服务器的拟态防御 Web 服务器

Fig.8 Compositions of the four test objects

图 8 4 种测试对象组成

利用 SprientAvalanche 31000b 测试仪仿真大量 HTTP 请求发送至被测服务器,测试图 7 中 4 种测试对象的新建速率、并发数、吞吐量和响应时间,结果见表 4.

Table 4 Testing records of performance

表 4 性能测试记录

测试对象	新建速率(/s)	并发数	吞吐量	响应时间(ms)
(a) 物理机对照	23 643	69 716	984 000	0.004
(b) 虚拟机对照	3 941	65 320	387 000	1.533
(c) 物理机拟态防御 Web 服务器	18 527	65 177	982 000	0.231
(d) 虚拟机拟态防御 Web 服务器	3 880	64 002	301 496	1.704

根据表 4 的结果,有如下结论.

- 对照情形(a)、情形(b)的测试数据可以发现:虚拟机服务器相对于物理机服务器在新建速率和吞吐量上有明显的降低,并发数也有少量降低,同时延长了响应时间约 1.5ms,说明虚拟机服务器代替物理机服务器在较大程度上降低了服务器系统的性能;
- 对照情形(c)、情形(d)的测试数据可以发现,与情形(a)、情形(b)的对照结论基本相同,进一步说明虚拟机服务器对系统性能的影响较为强烈;
- 对照情形(a)、情形(c)的测试数据可以发现,使用物理服务器作为变迁对象的拟态防御 Web 服务器在

新建速率和响应时间上的影响较为明显,说明拟态防御机制的使用有一定的性能代价;但在新建速率、并发数和吞吐量的数量级上没有降低;

- 对照情形(a)、情形(d)的测试数据可以发现:相比于普通 Web 服务器,使用虚拟机作为变迁对象的拟态防御 Web 服务器性能损失较大;然而结合前两条结论分析,可以判断出性能损失的主要来源是虚拟服务器的使用,拟态防御机制造成的性能损失并不是主要原因;
- 对照情形(a)、情形(c)和情形(b)、情形(d)的响应时间差值可以发现,两组对照的响应时间均存在约 0.2ms 的差值,可以推测,该时延差值是分发表决机制带来的固有时延,是拟态防御机制不可避免的时延.

综合以上结论,性能测试的结果表明:新建速率、并发数和吞吐量这 3 项指标在数量级上没有降低,分发表决机制存在约 0.2ms 的固有时延,因而拟态防御 Web 服务器相比同类服务器在性能上有一定的损失.由于性能测试对服务器系统的安全性和功能等方面没有要求对等,因而从性能单方面来看,拟态防御 Web 服务器是降低的.但是拟态防御 Web 服务器目前处于初步研制阶段,存在较大的优化空间.从测试数据中可以看出:物理机代替虚拟机就能够实现性能较大幅度的提高;分发表决算法的优化能够降低拟态防御机制的固有时延.因而在下一步工作中,拟态防御 Web 服务器面临着较大的优化工作量.

5.3 性能优化分析

拟态防御 Web 服务器作为具有“内生安全性”的系统设备,在附加安全工具、系统维护方面的成本整体低于采用传统防御技术的 Web 服务器系统.另外,通过改进拟态防御 Web 服务器的实现技术、优化 Web 服务器的性能,拟态防御 Web 服务器的安全性、功能和性能能够得到较大程度的提升.拟态防御 Web 服务器的优化工作主要包括拟态防御机制相关的实现技术优化和 Web 服务器的运行特性的优化.

针对拟态防御机制在 Web 服务器上的当前实现、分发表决模块以及服务器池的实现存在较大的优化空间.对于前置请求分发-响应表决模块,相比单台 Web 服务器,时延主要来源于该模块,可以通过分发表决模块功能最小化进行优化,以缩短固有时延.通过模块硬件化设计,仅保留分发、比较和表决等关键功能,形成请求分发-响应表决专用组件,以硬件实现代替软件实现,从而尽可能地缩短时延,提高效率,进而降低拟态防御 Web 服务器系统的整体时延.同时,该专用组件的形成,能够为其他拟态防御系统的实现提供高效的解决方案.对于基于虚拟机的 Web 服务器池,可以采用容器技术进行轻量化设计.利用容器承载 Web 应用,使 Web 服务以进程的形式运行于物理载体上,从而将运行时平台、操作系统等基础设施的安全性问题归约为容器安全,同时大大降低软/硬件资源消耗、增强可扩展性并增强拟态防御机制部署的灵活性.

针对拟态防御 Web 服务器的运行特性,由于拟态 Web 服务器在原型实现上侧重于对拟态防御原理的实现和安全性的验证,在对 Web 服务器的选取上,选择了简单搭建的 Web 服务器,对于系统的整体运行特性优化工作有所欠缺.依据传统的 Web 服务器优化方案[30],从整体上可分为硬件优化和软件优化.

- 硬件上,影响因素主要是内存、处理器、网络环境、硬盘等,根据服务器的实际使用场景,采用容量合适、可扩展性强的硬件是较好的选择;
- 软件上的优化又可分为应用的优化和服务器软件配置的优化:应用的优化,如网页静态化设计、建立动态内容缓存等,通过优化应用的设计,避免过多的响应时延和资源消耗;服务器软件的优化包括中间件配置的优化、服务器负载均衡、设置 url 缓存等,这些优化方法均能在降低响应时延、提高资源利用率上产生一定的效果.

另外,鉴于 Web 服务器系统与数据的紧密关联性,数据库的性能优化设计也能为 Web 服务器的性能提升带来益处.基于传统的 Web 服务器运行特性的优化方案,拟态防御 Web 服务器既要优化每个 Web 服务承载主体(执行体),也要优化动态选择机制,保证服务器池中冗余工作的 3 个执行体性能均衡且较优越,以避免“短板效应”.

6 总结与展望

拟态防御基于动态异构冗余结构,从系统结构层面内生安全性.通过异构性扰乱攻击的反馈信息,降低单步

攻击成功的概率.同时,通过动态性在时间维度上增大异构性,增大系统的不确定性,从而加大攻击难度.

相比于已有的典型防御技术,拟态防御在攻击链的各个阶段均具有较强的防御能力,能够应对针对已知和未知漏洞的攻击.

拟态防御 Web 服务器验证了拟态模型的可行性,同时说明拟态防御模型具有较好的发展前景.通过使用不同的异构技术、动态选择机制以及优化拟态防御机制的实现技术,可以构建高性价比的拟态防御 Web 服务器.拟态防御具备灵活的部署方式,能够适应不同的应用场景,在云安全、数据安全、系统安全等领域具有较大的发展空间.

本文分析了系统安全现状,并基于攻击链模型对已有安全技术进行了分析和对比,基于已有防御技术的分析,提出了拟态防御模型,并基于该模型构建实现了拟态防御 Web 服务器.通过安全性和性能测试,验证了拟态防御的有效性和可行性.最后,提出了拟态防御的发展前景和推广价值.

References:

[1] Internet Society of China, CNCERT/CC. China network sites developing situation and security report (2016). 2016 (in Chinese). http://tech.163.com/16/0320/15/BIK212JA00094P25.html

[2] Fang SW, Portante A, Husain MI. Moving target defense mechanisms in cyber-physical systems. In: Securing Cyber-Physical Systems. CRC Press, 2015. 63. https://books.glgoo.com/books?hl=zh-CN&lr=&id=wB6vCgAAQBAJ&oi=fnd&pg=PA63&ots=bkQgsF0K0T&sig=NMqbYCLX0YGm329DhO-0zLmxSIc#v=onepage&q&f=false

[3] Subrahmanian VS, Ovelgonne M, Dumitras T, Prakash BA. The Global Cyber-Vulnerability Report. Springer Int'l Publishing, 2015. 33−64. [doi: 10.1007/978-3-319-25760-0]

[4] China Information Technology Security Evaluation Center. China national vulnerability database of information security. 2015 (in Chinese). http://www.cnnvd.org.cn/vulnerability/statistics

[5] Xu H, Chen X, Zhou J, Wang Z. Research on basic problems of cognitive network intrusion prevention. In: Proc. of the 9th Int'l Conf. on Computational Intelligence and Security (CIS). 2013. 514−517. [doi: 10.1109/CIS.2013.114]

[6] Chung CJ, Khatkar P, Xing T, Lee J, Huang D. NICE: Network intrusion detection and countermeasure selection in virtual network systems. IEEE Trans. on Dependable and Secure Computing, 2013,10(4):198−211. [doi: 10.1109/TDSC.2013.8]

[7] Madan BB, Goševa-Popstojanova K, Vaidyanathan K, Trivedi KS. A method for modeling and quantifying the security attributes of intrusion tolerant systems. Performance Evaluation, 2004,56(1-4):167−186. [doi: 10.1016/j.peva.2003.07.008]

[8] Okhravi H, Hobson T, Bigelow D, Streilein W. Finding focus in the blur of moving-target techniques. Security & Privacy, 2014,12(2):16−26. [doi: 10.1109/MSP.2013.137]

[9] Vasilomanolakis E, Karuppayah S, User M, Fischer M. Taxonomy and survey of collaborative intrusion detection. ACM Computing Surveys (CSUR), 2015,47(4):55. [doi: 10.1145/2716260]

[10] Liao HJ, Lin CHR, Lin YC, Tung KY. Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 2013,36(1):16−24. [doi: 10.1016/j.jnca.2012.09.004]

[11] Whitea JS, Fitzsimmonsb T, Matthewsc JN. Quantitative analysis of intrusion detection systems: Snort and suricata. Proc. of the SPIE, 2013,8757:875704-1. [doi: 10.1117/12.2015616]

[12] Kenkre PS, Pai A, Colaco L. Real time intrusion detection and prevention system. In: Proc. of the 3rd Int'l Conf. on Frontiers of Intelligent Computing: Theory and Applications (FICTA 2014). Springer Int'l Publishing, 2015. 405−411. [doi: 10.1007/978-3-319-11933-5_44]

[13] Ho CY, Lai YC, Chen IW, Wang FY, Tai WH. Statistical analysis of false positives and false negatives from real traffic with intrusion detection/prevention systems. IEEE Communications Magazine, 2012,50:146−154. [doi: 10.1109/MCOM.2012.6163595]

[14] Song J, Takakura H, Okabe Y, Nakao K. Toward a more practical unsupervised anomaly detection system. Information Sciences, 2013,231:4−14. [doi: 10.1016/j.ins.2011.08.011]

[15] Vaidya N, Godbole P. Hardware implementation of key functionalities of NIPS for high speed network. In: Proc. of the Computing and Network Communications. 2015. 892−897. [doi: 10.1109/CoCoNet.2015.7411296]

[16] Wang F, Uppalli R, Killian C. Analysis of techniques for building intrusion tolerant server systems. In: Proc. of the Military Communications Conf., Vol. 2. 2003. 729−734. [doi: 10.1109/MILCOM.2003.1290202]

[17] Powell D, Stroud R. Conceptual model and architecture of MAFTIA. Technical Report, University of Newcastle Upon Tyne Computing Science, 2003. 23−29.

[18] Wang F, Jou F, Gong F, Sargor C, Gosevapopstojanova K. SITAR: A scalable intrusion-tolerant architecture for distributed services. In: Proc. of the Workshop on Information Assurance and Security. 2003. 38−45. [doi: 10.1109/DISCEX.2003.1194957]

[19] Nguyen QL, Sood A. A comparison of intrusion-tolerant system architectures. IEEE Security & Privacy, 2011,9(4):24−31. [doi: 10.1109/MSP.2010.145]

[20] Yu J, Cheng XG, Li FG, Pan ZK, Kong FY, Hao R. Provably secure intrusion-resilient public-key encryption scheme in the standard model. Ruan Jian Xue Bao/Journal of Software, 2013,24(2):266−278 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4324.htm [doi: 10.3724/SP.J.1001.2013.04324]

[21] Zhang XY, Li ZB. Overview on moving target defense technology. Communications Technology, 2013,46(6):111−113 (in Chinese with English abstract).

[22] Antonatos S, Akritidis P, Markatos EP, Anagnostakis KG. Defending against hitlist worms using network address space randomization. Computer Networks, 2007,51(12):3471−3490. [doi: 10.1016/j.comnet.2007.02.006]

[23] Huang Y, Ghosh A. Automating intrusion response via virtualization for realizing uninterruptible Web services. In: Proc. of the Network Computing and Applications (NCA 2009). 2009. 114−117. [doi: 10.1109/NCA.2009.37]

[24] Shacham H, Page M, Pfaff B, Goh E-J, Modadugu N, Boneh D. On the effectiveness of address-space randomization. In: Proc. of the 11th ACM Conf. on Computer and Communications Security. 2004. 298−307. [doi: 10.1145/1030083.1030124]

[25] Salamat AG, Franz M. Reverse stack execution in a multivariant execution environment. In: Proc. of the Workshop Compiler and Architectural Techniques for Application Reliability and Security. 2008. 1−7. http://babaks.com/files/catars08.pdf

[26] Nguyentuong A, Evans D, Knight JC, Cox B, Davidson JW. Security through redundant data diversity. In: Proc. of the IEEE Int'l Conf. on Dependable Systems and Networks with FTCS and DCC (DSN 2008). 2008. 187−196. [doi: 10.1109/DSN.2008.4630087]

[27] Huang Y, Ghosh AK. Introducing diversity and uncertainty to create moving attack surfaces for Web services. In: Proc. of the Moving Target Defense. New York: Springer-Verlag, 2011. 131−159. [doi: 10.1007/978-1-4614-0977-9_8]

[28] Okhravi H, Rabe MA, Mayberry TJ, Leonard WG, Hobson TR, Bigelow D, Streilein WW. Survey of cyber moving targets. Technical Report, No. MIT/LL-TR-1166, Massachusetts Inst of Technology Lexington Lincoln Laboratory, 2013.

[29] Wang ZY, Yang XJ, Zhou Y. Scalable triple modular redundancy fault tolerance mechanism for MPI-oriented large scale parallel computing. Ruan Jian Xue Bao/Journal of Software, 2012,23(4):1022−1035 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4011.htm [doi: 10.3724/SP.J.1001.2012.04011]

[30] Wang YN, Wu HR, Huang F. Optimization analysis and research of high concurrency Web application system performance. Computer Engineering and Design, 2014,35(8):2976−2980 (in Chinese with English abstract).

附中文参考文献:

[1] 中国互联网站发展状况及其安全报告(2016).2016. http://tech.163.com/16/0320/15/BIK212JA00094P25.html

[4] 中国国家信息安全漏洞库.2015. http://www.cnnvd.org.cn/vulnerability/statistics

[20] 于佳,程相国,李发根,潘振宽,孔凡玉,郝蓉.标准模型下可证明安全的入侵容忍公钥加密方案.软件学报,2013,24(2):266−278. http://www.jos.org.cn/1000-9825/4324.htm [doi: 10.3724/SP.J.1001.2013.04324]

[21] 张晓玉,李振邦.移动目标防御技术综述.通信技术,2013,46(6):111−113.

[29] 王之元,杨学军,周云.大规模MPI并行计算的可扩展三模冗余容错机制.软件学报,2012,23(4):1022−1035. http://www.jos.org.cn/1000-9825/4011.htm [doi: 10.3724/SP.J.1001.2012.04011]

[30] 王亚楠,吴华瑞,黄锋.高并发Web应用系统的性能优化分析与研究.计算机工程与设计,2014,35(8):2976−2980.

仝青(1992－),女,河南郑州人,学士,主要研究领域为网络空间安全.

张为华(1974－),男,博士,副教授,CCF 专业会员,主要研究领域为计算机体系结构,软件纠错,编译器优化.

张铮(1976－),男,博士,副教授,主要研究领域为网络空间安全.

邬江兴(1953－),男,教授,博士生导师,主要研究领域为信息通信网络,网络空间安全.