端信息跳变系统自适应策略研究

赵春蕾[1,2,3]，贾春福[1]，翁臣[1]，林楷[1]

（1. 南开大学 信息技术科学学院，天津 300071；2. 天津理工大学 计算机视觉与系统省部共建教育部重点实验室，天津 300384；
3. 天津理工大学 智能计算及软件新技术天津市重点实验室，天津 300384）

摘 要：提出自适应端信息跳变策略，在原有跳变系统中增加攻击检测、反馈传输和自适应控制模块，实时分析系统受攻击情况，调整跳变策略，以降低系统踏入攻击者目标范围的概率。实验验证了模型系统的抗攻击性，证明了端信息跳变系统中自适应策略的可行性与有效性。

关键词：网络安全；主动防御；端信息跳变；自适应策略；代理

中图分类号：TP393.08 **文献标识码**：A **文章编号**：1000-436X(2011)11A-0051-07

Research on adaptive strategy for end-hopping system

ZHAO Chun-lei[1,2,3], JIA Chun-fu[1], WENG Chen[1], LIN Kai[1]

(1. College of Information Technical Science, Nankai University, Tianjin 300071, China;
2. Key Laboratory of Computer Vision and System, Ministry of Education, Tianjin University of Technology, Tianjin 300384, China;
3. Tianjin Key Lab of Intelligent Computing & Novel Software Technology, Tianjin University of Technology, Tianjin 300384, China)

Abstract: An adaptive strategy was proposed for EH system. Modules of attack detection, feedback transmission and adaptive control were added to original EH system. They could help to make threats analysis and adjust hopping strategy in real time, so as to reduce the probability of the system stepping into the attackers' target area. With the experiments proving the anti-attack ability of the prototype system, the feasibility and effectiveness of adaptive strategy were demonstrated for EH system.

Key words: network security; active defense; end-hopping; adaptive strategy; agent

1 引言

端信息跳变（EH, end-hopping）技术是一种主动网络防御技术。在端到端的通信中，或端节点内部不同设备和子系统之间的数据传输中，通信双方按照一定的跳变方案，伪随机改变一方或双方的端信息(EI, endpoint information)，从而达到迷惑、干扰敌手，保护网络通信和数据安全的目的[1]。

端信息跳变技术对传统的大规模网络攻击具有很好的防御效果[2]。目前国内外对端信息跳变技术的研究成果主要集中在 2 个方面：一个是关于跳变内容的研究，出现了如端口跳变[3]、地址跳变[4,5]、协议跳变[6]和混合跳变等技术；另一个是关于同步机制的研究，出现了严格时间同步机制[3]，基于 ACK 报文的同步机制[7]，时间戳同步机制[8]以及分布式同步机制[9]等。

收稿日期：2011-08-24

基金项目：国家自然科学基金资助项目（60973141）；天津市自然科学基金资助项目（09JCYBJ00300）；高等学校博士学科点专项科研基金资助项目（20100031110030）；网络安全与密码技术福建省高校重点实验室开放课题（2011004）

Foundation Items: The National Natural Science Foundation of China (60973141); The Natural Science Foundation of Tianjin (09JCYBJ00300); The Specialized Research Fund for the Doctoral Program of Higher Education of China (20100031110030); Funds of Key Lab of Fujian Province University Network Security and Cryptology(2011004)

随着网络攻击技术的发展，目前攻击者在发起有效攻击前往往先进行扫描和踩点[10]，利用所获取的信息帮助进行目标定位，缩窄攻击范围。本文中的实验表明，对于一个采用伪随机序列进行匀速跳变的端信息跳变系统，如果攻击者进行有指导的选择攻击（即，首先分析和估计跳变系统的所有可用跳变节点，然后选取其中部分节点进行大强度攻击），当攻击者破坏跳变系统 40%以上可用跳变节点时，整个系统的服务性能将遭受严重影响，甚至导致服务失败。

针对上述问题，本文在对端信息跳变技术进行深入研究的基础上，将自适应技术与端信息跳变技术相结合，提出自适应端信息跳变策略，在原有跳变系统中增加攻击检测、反馈传输和自适应控制模块，通过对各跳变节点上受攻击情况进行实时评估，指导下一跳选择方案，将通信质量恶劣的节点及时从可用跳变节点集中剔除，从而避开干扰或攻击，提高网络传输的质量，使服务性能得到较大改善。

2 模型和定义

为方便描述，以端信息跳变系统中的 IP 地址跳变为基础进行讨论，所得自适应策略可扩展至端口跳变、协议跳变以及混合跳变。假设攻击者不能阻塞网络，攻击者的攻击数据分组在网络中无丢失，且数据分组在网络信道外无损失。这些前提和假设能使分析集中在攻击和防御的策略上。本文不关心攻击方在具体受控攻击点上的部署问题，只关心受攻击方每个 IP 地址上接收数据分组的情况。

2.1 针对型攻击模型

在端信息跳变系统中，将每个可用 IP 地址定义为一个节点(node)，所有可用跳变节点的集合为 *Node*。设跳变系统全部可用跳变节点数量为 r，即

$$Length(Node)=r$$

任意时刻，端信息跳变系统中仅有一个节点提供数据传输服务，称为活动节点，其他节点均为非活动节点。ρ 为活动节点选择参数。

$$\rho_i=\begin{cases}1, & node_i\text{为活动节点}\\ 0, & node_i\text{为非活动节点}\end{cases},\quad i\in\{1,2,\cdots,r\}$$

$$\sum_{i=1}^{r}\rho_i=1$$

攻击者的最大攻击能量用参数 *Nrg* 表示，*Nrg* 为定值，用来限定攻击者的能力，这是讨论防御策略的前提。*Nrg* 用单位时间内，攻击者可发送到网络中的数据分组的最大数量来描述。

攻击者在可用跳变节点集中选择的攻击目标用 *Target* 表示，则

$$Target\subset Node$$

第 i 个节点上所承受的攻击强度为 *Nrg*，则

$$Nrg\geqslant\sum_{node_i\in Target}Nrg_{node_i}$$

假设敌手能窃听网络中的数据分组，分析端信息跳变系统使用的 IP 地址范围，用所得到的知识调整下一步进攻方案，并将此方案分发、部署到各受控攻击节点上，但这些操作需要相当长的时间。设自上次使用目标节点集 *Target* 到调整方案后使用 *Target'*进行攻击至少需要 t_a 的时间。

用下述模型进行描述：

Capture: ch_cap (packet)

IP(packet) →$Target_{temp}$

Analysis: if($Length(Target_{temp})$=1)

Then $Target= Target_{temp}$

else $Target$=Ltd($Target_{temp}$)

Deploy: $Target$→ controlled nodes

Attack: Atk_packet→$Target$

攻击者遵守下述原则：总是尽量缩小攻击范围，以使攻击目标 *Target* 上的攻击强度更大。当发现服务地址固定时，所有能量集中于该地址进行攻击；当发现系统采用端信息跳变技术服务时，选择收发包数量多的 IP 地址附近一个地址段进行攻击。

2.2 端信息跳变系统模型

在端信息跳变系统中，使用跳代理技术[1]进行部署，实现匀速伪随机地址跳变。客户端(client)、服务器(Web server)通过跳代理(hopping proxy)完成伪随机跳变通信。跳代理由控制中心(control center)、可用跳变节点集 *Node* 以及转发节点 *F* 组成。每个可用跳变节点部署一个 IP 地址，在任意时刻，仅有一个节点处于活动状态。合法客户端发送的数据分组，通过当前活动节点 $node_i$ 进入跳变代理，经转发节点 *F* 发送给服务器。服务器发出的数据分组经转发节点和活动节点返回给客户端。在控制中心的统一控制下，完成活动节点的切换以及同步工作。系统结构如图 1 所示。

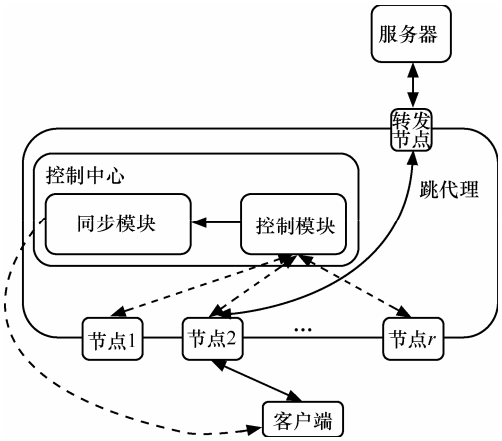


图 1　端信息跳变系统结构

跳变系统使用的跳变算法为 $f(t, key)$，其中 f 为伪随机函数，t 为当前时间参数，key 为客户端、服务器之间的共享密钥。当前时间 t 和收发双方约定的密钥 key 唯一确定了一个 IP 地址，即下一跳地址。

$$IP_{\text{next}} = f(t, key)$$

在准备阶段，控制中心设定同步信息 Syn，将其发布给各客户端和可用跳变节点。如下所示：

Control Center→Client:　key

Control Center:　　//generate Syn information

Syn={ $f(t, key)$, t_{start}, t_{stop}, t_{onehop}, $Clock$}

Control Center→Node:　Syn

//transmit Syn to Node

Control Center→Client: Syn

//transmit Syn to Client

准备阶段结束后，跳代理与客户端之间完成跳变算法 $f(t, key)$、跳变服务开始时间 t_{start}、结束时间 t_{stop}、一跳持续时间 t_{onehop}，以及时钟 $Clock$ 的同步。当时间到达 t_{start}，端信息跳变系统开始协同工作。

开始端信息跳变信息传输服务后，跳代理和客户端均定期与时间服务器同步校准本地时钟。整个服务时间内，真正的服务器隐藏在跳代理之后，其 IP 地址为固定值，在服务过程中与普通 Web 服务器无区别。跳代理使用伪随机函数 $f(t, key)$来决定下一跳地址。活动节点在客户端与服务器之间转发数据分组。非活动节点收到数据分组时，进行丢弃处理。客户端使用相同伪随机函数计算当前服务 IP，与服务器进行数据传输。

客户端发送数据分组给服务器的过程用如下模型描述：

Control Center:　　for　$i \in (1,2,\cdots,r)$ do

　　if IP($node_i$)= $f(t, key)$ then set $node_i$ ACTIVE

　　else set $node_i$ SLEEP

Client:　　IP= $f(t, key)$, creat(packet), send (packet)

$node_i$ on ACTIVE:　on ch_recv (packet)　send (packet) to F

$node_i$ on SLEEP:　on ch_recv (packet)　drop (packet)

F:　on ch_recv (packet) send(packet) to Web Server

Web Server:　　recv(packet) from F

从服务器发送数据分组给客户端的过程与之类似，这里不再赘述。

2.3　自适应策略下的防御模型

2.3.1　受攻击情况估计

在端信息跳变系统中，可将非服务节点收到的数据分组(AllPacket)分为 3 种类型。

1) 无关分组(NRltPacket)，其他主机进行网络通信而发送到网络中的一些无关的数据分组。

2) 不同步分组(USynPacket)，跳变系统在下一跳切换过程中，由于边界同步失败产生的少量不同步分组。

3) 攻击分组(AtkPacket)，攻击者针对本节点进行攻击产生的数据分组。

经实验测试，前 2 种数据分组流量小且较为稳定，而攻击分组流量起伏较大，当有攻击发生时，攻击分组的数量级远大于前 2 种数据分组，即

$$Qty(AllPackets) = Qty(NRltPackets) + Qty(USynPackets) + Qty(AtkPackets) \approx Qty(AtkPackets)$$

因此，可用非服务节点所接收的数据分组数量 $Qty(AllPackets)$ 来估算该节点接收攻击分组的数量 $Qty(AtkPackets)$，以此评估受攻击情况，帮助调整跳变策略。

2.3.2　自适应策略下的防御模型

在原有简单跳变系统中，为每个可用跳变节点增加代理模块 Agent，用于实时检测并记录当前非活动节点的受攻击情况，定期反馈给控制中心。系统结构如图 2 所示。

控制中心统计所有非服务节点受攻击情况，重新制定跳变策略，并在 t_d 时间内分发给各节点及客户端。各节点及客户端按新策略重新选择下一跳地址，以躲避攻击。后续实验证明，当 t_a 不小于 $5t_d$

时，自适应跳策略能达到较好的防护效果。为增加自适应特性，需对原计算 IP 地址的伪随机函数进行调整，

$$IP_{\text{next}} = f(t, key, List)\ ,\ List_{\text{start}} = Node$$

其中，*List* 为当前可用跳变节点列表，$List_{\text{start}}$ 为初始可用跳变节点。利用函数 *f*(*t*, *key*)在 *List* 中伪随机选择下一跳地址。

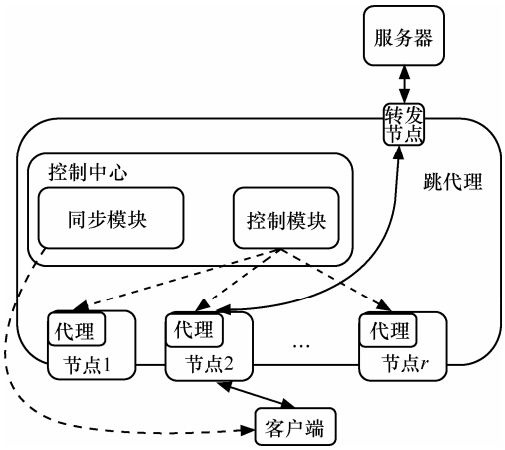


图 2 自适应端信息跳变系统结构

在准备阶段，控制中心设定同步信息 Syn，将其发布给各客户端和各可用跳变节点。如下所示：

Control Center→Client: (*key*, $List_{\text{start}}$)

Control Center: Syn={ *f(t, key, list)*, t_{start}, t_{stop}, t_{onehop}, *Clock*}

//generate Syn information

Control Center→Node: Syn //transmit Syn to Node

Control Center→Client: Syn //transmit Syn to Client

准备阶段结束后，跳代理与客户端之间完成跳变算法 *f*(*t*, *key*, *List*)、跳变开始时间 t_{start}、结束时间 t_{stop}、一跳持续时间 t_{onehop}，以及时钟 *Clock* 的同步。当时间到达 t_{start}，端信息跳变系统开始协同工作。

开始端信息跳变信息传输服务后，跳代理和客户端均定期校准本地时钟，与时间服务器同步，控制中心根据所收集的当前各非服务节点受攻击情况进行分析判断，改变跳变策略，制定下一跳方案，分发至各跳变节点和客户端，此过程循环。

数据传输过程与原端信息跳变系统相比，仅需将跳变函数由 *f*(*t*, *key*)改为 *f*(*t*, *key*, *List*)。重点描述非活动节点受攻击情况统计及控制中心的策略调整过程。设定开关参数 ω，当单位时间内收到的数据分组数量大于 ω 时认为有攻击发生，ω 的取值由 3.1 节实验得出。非活动节点上的代理每收到一个新数据分组，将计数器 *Counter* 加 1。在一个计数周期结束时，将受攻击强度大于 ω 的节点上报控制中心，控制中心从 *List* 中把受攻击节点去除，使用伪随机函数 *f*(*t*, *key*, *List*)生成下一跳地址。可以用下述过程描述：

Agent on $node_i$ ($node_i$ is SLEEP):

Timer= ϕt_{onehop}, $Counter_i$=0

while (Timer)

on ch_recv (packet) $Counter_i = Counter_i$++

if $Counter_i > \omega$ then report ($node_i$) to Control Center

Control Center: $List = List_{start} - (node_i)_{\text{reported}}$

send (*List*) to Client

Client: IP = *f*(*t*, *key*,*list*)

其中计数周期（Timer）为一跳持续时间（t_{onehop}）的 ϕ 倍，较小的 ϕ 值可以保证攻击发现的实时性，但会带来频繁的反馈传输使系统效率降低，实验证明 ϕ 值取 5 时防御效果最好，因此在 3.2 节中取 ϕ =5 进行实验。

自适应策略的核心在于，对当前各节点受攻击情况进行实时分析，选择未受攻击的节点作为下一跳地址。优点是可用节点集会随受攻击情况自适应变化，在一定程度上躲避敌手攻击。但此方案也有不足之处，即当敌手能从长时间观测中分析推断跳变策略，可以通过精心设计的攻击包来主动压缩可用跳变节点集，将下一跳地址引导至其预期节点，以此作为目标进行下一轮攻击。为防止这种情况发生，需对本策略进行改进。改进的策略如下：

Agent on $node_i$ ($node_i$ is SLEEP):

Timer= ϕt_{onehop}, $Counter_i$=0

while (Timer)

on ch_recv (packet) $Counter_i = Counter_i$++

if $Counter_i > \omega$ then report ($node_i$) to Control Center

Control Center: $List = List_{\text{start}} - \text{select}((\text{node}_i)_{\text{reported}}, \mu)$

send (*List*) to Client

Client: IP = *f* (*t*, *key*, *List*)

select 描述了这样的动作，当上报的节点数不超过总可用节点数的 μ%时，去除所有受攻击节点；当上报的节点数超过总可用节点数的 μ%时，以 μ%的比例在 $List_{start}$ 中去除受攻击强度最大的节点，如果受攻击强度无差异，则随机选取其中 μ%的受攻击节点，在可用节点中去除。3.2 节中以 30 作为 μ 的值，保证系统在最坏情况下仍有 70%的节点可用。参数 μ 的加入保证了系统既具有一定的自适应性，又不失去端信息跳变的随机特性。

改进后的自适应策略能大大降低跳变系统踏入敌手攻击目标范围的可能性，很好地躲避敌手进行的选择攻击。而且，当敌手的攻击变成全地址攻击时，此策略也自动退化为匀速简单跳变，即当所有节点所受攻击相同时，地址选择函数将以等概率从所有可用地址中选择下一跳节点。

3　实验数据分析

在原型系统上进行了攻击实验，实验环境配置如表 1 所示。

表 1　　实验环境配置

实验模块	操作系统	网络带宽（Mbit/s）	IP 地址	数量	功能
服务器	Red Hat Linux 9	1000	12.12.12.4	1	Web
客户机	Win 2003	1000	13.13.14.4-13.13.14.7	4	
攻击者	Win2003	1000	13.13.14.8-13.13.14.12	5	CC 攻击
同步模块	Ubuntu 10	1000	13.13.14.1	1	UDP
控制模块	Ubuntu 10	1000	12.12.12.241	1	TCP
跳变节点	Ubuntu 10	1000	12.12.12.231-12.12.12.240	10	TCP

3.1　全地址攻击实验

采用无跳变技术保护的 Web 服务和地址池中有 10 个可用地址的端信息跳变系统进行实验。攻击方针对全部可用地址进行攻击，攻击强度逐渐增大。分别对非跳变、简单跳变、自适应跳变 3 种系统进行攻击，记录访问请求的平均响应时延，其结果如图 3 所示。

未受保护的服务器在遭受 CC(challenge collapsar)攻击时，随攻击强度增大，响应时延逐步增大。当攻击强度达到 30packet/s 时，响应时延迅速增大，超过 30packet/s 时已无法提供正常服务，此时表现为客户端无法打开服务器网页。由此可知，服务节点在攻击强度达到 30packet/s 时无法提供正常服务。针对端信息跳变系统进行攻击时，当某一可用跳变节点上的攻击强度大于 30packet/s，该节点被淹没，可视为不可用。因此可将 30packet/s 作为实验中部署选择性攻击时攻击强度的指标，即当针对某一节点进行攻击时，对其攻击强度不小于 30packet/s。同时，30packet/s 可作为判断该节点受攻击的开关阈值，也就是在 2.3.2 节中所使用的 ω。

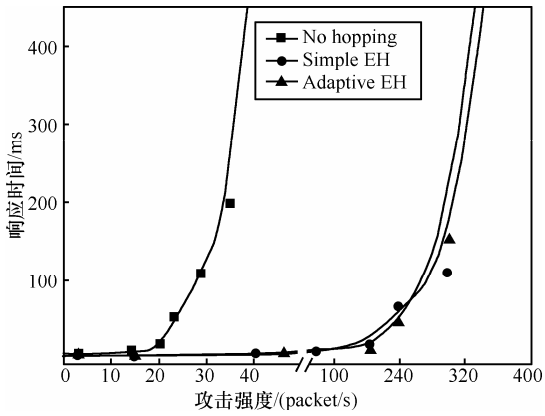


图 3　全地址攻击实验

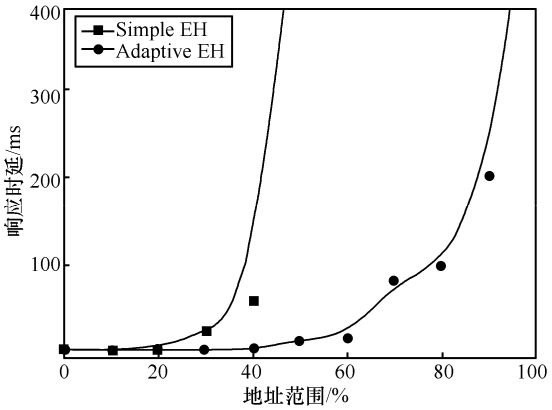
观察图 3 可得到如下结论：CC 全地址攻击下，未使用跳变技术保护的服务在攻击强度大于 30packet/s 时无法提供服务；采用简单端信息跳变技术保护的服务，在攻击强度大于 300packet/s 时响应时延开始增大，此时每个可用跳变节点上所受攻击强度均超过 30 packet/s；采用自适应端信息跳变技术保护的服务与采用简单端信息跳变技术保护的服务差别不大，这是因为，当所有可用地址所受攻击强度相同时，自适应端信息跳变系统退化为简单端信息跳变系统。

3.2　选择攻击实验

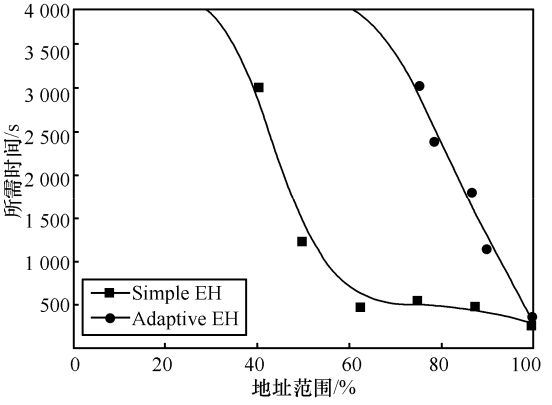
本文采用地址池中有 10 个可用地址的端信息跳变系统进行实验，一组使用匀速不变的简单跳变策略，另一组采用改进的自适应跳变策略进行对比。

总攻击能量一定时，攻击者选择一定百分比的可用跳变节点进行攻击。设总攻击能量大于 300packet/s，此种情况下，即使攻击范围增大至可用地址的 100%，也能使分配到每个目标地址的攻击强度大于 30packet/s，保证落入攻击范围内的地址承受 30packet/s 以上的攻击强度，这种情况对敌手最有利。

图 4(a)描述了节点受攻击范围与服务响应时延的关系。由图可知，端信息跳变系统在遭受有指导的洪泛攻击时，简单跳变系统当受攻击节点比例达到 30%的时候，响应时延迅速增大，当超过 40%时，响应时延趋于无穷大，发生拒绝服务。而自适应端信息跳变系统的响应时延增长较平缓，原因是通过从可用跳变节点集中去掉受攻击节点，达到一定躲避攻击的效果。而当受攻击节点比例继续增大时，自适应跳系统的响应时延也随之增大，当攻击范围接近全部可用地址的 100%时，CC 攻击退化为全地址攻击，自适应跳系统也随之退化为简单端信息跳变系统，由于此时所有可用跳变节点均遭受超过 30 packet/s 的攻击强度，因此自适应系统的响应时延也随之迅速增大。



(a) 攻击范围与服务响应时延的关系



(b) 攻击范围与达到拒绝服务所需时间的关系

图 4 选择攻击实验

图 4(b)描述了受攻击范围与达到拒绝服务所需时间的关系。随着攻击范围逐步增大，攻击者成功通过服务节点到达 Web 服务器的攻击分组也随之增多，从而达到拒绝服务所用的时间也逐渐缩短。简单跳变系统下，攻击范围在超过 40%以后首次出现拒绝服务，但此时需要持续攻击 1 300s 的时间才能出现拒绝服务。随攻击范围逐步增大，攻击者达到拒绝服务所需的攻击时间逐步缩短，当攻击范围接近 100%时，只需 290s 时间可使系统达到拒绝服务。而自适应跳变系统下情况较好，仅当攻击范围超过 80%时，首次出现拒绝服务，而当攻击范围接近 100%时，CC 攻击退化为全地址攻击，自适应跳系统也随之退化为简单端信息跳变系统，因此防御效果相同。

由此可知，应用了自适应策略的端信息跳变系统，其抗选择性攻击的效果远远优于简单端信息跳变系统。

4 结束语

本文在对端信息跳变技术进行深入研究的基础上，将自适应技术与端信息跳变技术结合起来，提出自适应端信息跳变策略。分别为攻击者、端信息跳变系统和自适应跳变系统建立了模型，并对自适应跳变策略的关键问题进行了探讨。在此基础上，建立原型系统并进行攻击实验。实验结果表明，自适应跳变策略能大大改善端信息跳变系统在选择攻击下的抗攻击性能。

参考文献：

[1] ZHAO C L, JIA C F, LIN K. Technique and application of End-hopping in network defense[A]. 1st ACIS International Symposium on Cryptography, and Network Security, Data Mining and Knowledge Discovery, E-Commerce and Its Applications, and Embedded Systems, CDEE 2010[C].2010.266-270.

[2] SHI L Y, JIA C F, LÜ S W, *et al*. Port and address hopping for active cyber-defense[A]. Pacific Asia Workshop on Intelligenceand Security Informatics[C]. 2007.295-300.

[3] LEE H, THING V. Port hopping for resilient networks[A]. 60th IEEE Vehicular Technology Conference[C]. 2004.3291-3295.

[4] SIFALAKIS M, SCHMID S, HUTCHISON D. Network address hopping: a mechanism to enhance data protection for packet communications[A]. IEEE International Conference on Communications[C].2005.1518-1523.

[5] ATIGHETCHI M, PAL P P, JONES C C, *et al*. Building auto-adaptive distributed applications: the quo-apod experience[A]. 23rd Interna-

tional Conference on Distributed Computing Systems Workshops[C]. 2003. 104-109

[6] 李树军. 基于协议转变的拒绝服务攻击技术的研究[J]. 计算机应用, 2006, 26(10): 2323-2325.
LI S J. Research on technology of DoS based on protocol transform [J]. Computer Applications, 2006, 26(10): 2323-2325.

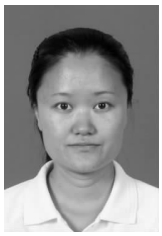
[7] BADISHI G, HERZBERG A, IDIT K. Keeping denial-of-service attackers in the dark[A]. IEEE Transactions on Dependable and Secure Computing[C]. 2007. 191-204.

[8] 石乐义. 基于端信息跳变的主动式网络防御策略研究[D]. 天津:南开大学, 2008.
SHI L Y. Research on the End Information Hopping Mechanism for Proactive Cyber Defense [D]. TianJin: Nankai University, 2008.

[9] LIN K, JIA C F, WENG C. Distributed timestamp synchronization for end hopping[J]. China Communications, 2011, 8(4): 164-169.

[10] 诸葛建伟，叶志远，邹维. 攻击技术分类研究[J]. 计算机工程. 2005, 21(21): 121-123.
ZHUGE J W, YE Z Y, ZHOU W. Research on classification of attack technologies [J]. Computer Engineering, 2005, 21(21): 121-123.

作者简介：



赵春蕾（1979-），女，河北唐山人，南开大学博士生，天津理工大学教师，主要研究方向为网络与信息安全。

贾春福（1967-），男，河北文安人，博士，南开大学教授、博士生导师，主要研究方向为信息安全与可信计算、恶意代码发现与分析。

翁臣（1986-），男，湖北恩施人，南开大学硕士生，主要研究方向为网络安全。

林楷（1986-），男，江西上饶人，南开大学博士生，主要研究方向为网络和信息安全、信息内容安全。