# 环形结构的入侵容忍系统研究

周 华[1,2]，孟相如[2]，张 立[2]，乔向东[2]

(1. 西安通信学院通信装备管理系 西安 710106; 2. 空军工程大学电讯工程学院 西安 710077)

**【摘要】**提出了一种环形的入侵容忍系统结构，并描述了数据一致性和服务器控制算法。当在线服务器数目大于两倍故障服务器数目时，该系统实现了服务数据的正确一致性，并通过动态调整在线服务器数目提高了系统的入侵容忍能力。仿真实验分析了在面临入侵情况下系统的可用性性能，并在实例中得出了该系统可用性较优的时间区间，为进一步提高系统的安全性和服务能力提供了一定的依据。

**关 键 词** 环形结构; 可用性; 一致性; 入侵容忍; 网络安全

**中图分类号** TP309.2 **文献标识码** A doi:10.3969/j.issn.1001-0548.2010.04.025

# Annular Architecture for the Intrusion-Tolerance System

ZHOU Hua[1,2], MENG Xiang-ru[2], ZHANG Li[2], and QIAO Xiang-dong[2]

(1. Department of Communication Equipment Management, Xi'an Communication Institute Xi'an 710106;

2. Telecommunication Engineering Institute, Air Force Engineering University Xi'an 710077)

**Abstract** An annular architecture for the intrusion-tolerance system is presented. The data consistency algorithm and control algorithm for servers are described in detail. This system guarantees the data consistency when the total of servers online is more than twice the number of malicious servers, and the performance of intrusion-tolerance can be improved by adjusting the number of servers online dynamically. The system availability is illustrated in the simulation experiments and a time zone of higher availability is also presented.

**Key words** annular architecture; availability; consistency; intrusion tolerance; network security

随着计算机的普及和信息技术的迅速发展，人们对计算机网络系统的安全性要求愈来愈高。传统的网络安全技术已不能完全保护网络系统不受攻击，因此产生了一种新的安全措施——入侵容忍[1]。入侵容忍的主要思想是承认系统中存在可以被攻击者利用的脆弱点，并构建一种可以容忍一定数量故障(包括攻击和入侵)的系统[2-3]。

本文提出了一种基于可信实时计算基(TTCB)[4-6]的环形入侵容忍系统。与相关研究工作[7-9]不同，该系统采用中央控制器控制服务器的在线和离线状态，各个服务器轮流在线提供服务的工作方式。系统在最大故障服务器数目为 $f$ 的条件下，只要在线服务器数目不小于 $2f+1$，即可满足系统对客户端服务数据的正确一致性，达到入侵容忍的目的。通过仿真实验分析了在发生入侵的情况下，入侵容忍系统的可用性性能。

## 1 环形入侵容忍系统的体系结构

环形入侵容忍系统主要由客户端、服务器、中央控制器等组成。中央控制器控制服务器的在线与离线状态。客户端访问服务器及服务器之间的通信通过负载网络进行。负载网络采用点对点的通信方式，并且通信链路是可靠的[7]。除了服务器的本地安全内核——TTCB，系统的时间模型总体上是异步的。假设入侵容忍系统中正确的服务器严格按照协议规范执行，而发生入侵的服务器会出现拜占庭故障，即故障服务器的行为是任意的，它们可以篡改、伪造数据，无限地延迟提交数据，停止执行协议，甚至合谋产生错误的结果等。环形入侵容忍系统的结构如图1所示。

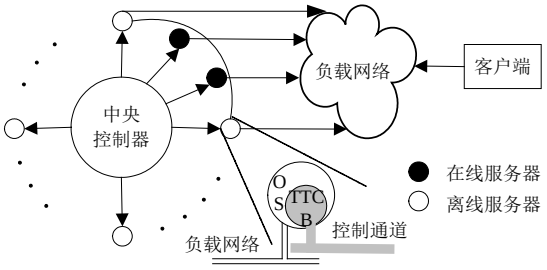



图1 环形入侵容忍系统的体系结构

收稿日期：2009－04－07; 修回日期：2009－09－20

基金项目：国家自然科学基金(60774091)；陕西省自然科学基金(SJ08F14)

作者简介：周 华(1981－)，男，博士生，主要从事计算机网络安全方面的研究.

### 1.1 TTCB

TTCB是一种安全可信的实时分布式模块，它植入服务器内部并与操作系统隔离。每个服务器内部的TTCB称为本地TTCB，所有本地TTCB通过专有的控制通道进行连接。任何应用进程都不能破坏TTCB的正确运行，而TTCB只会以停止工作的方式发生故障。每个本地TTCB拥有一个时钟，并且所有的时钟都是同步的。

TTCB通过接口向进程提供有限的服务，主要分为安全服务和时间服务[4]两类。入侵容忍系统使用了以下TTCB的3个服务：

(1) 本地认证服务(LA)。该服务允许进程与本地TTCB安全通信，服务在进程运行之前认证本地TTCB，并为TTCB与进程建立一个共享的对称密钥。该密钥可以作为两者之间的安全通道，用来加密和认证它们之间的通信。

(2) 可信块协商服务(TBA)。该服务是安全协议的主要组成部分，在对一组进程提议的值协商后，该服务就将达成一致的某一个值提交给进程。可信块协商服务主要由TTCB_propose、TTCB_decide以及decision 3个函数实现。

(3) 可信绝对时间戳服务(TAT)。该服务提供具有全局意义的时间戳。

客户端向在线服务器发送请求消息后，各个在线服务器分别将处理结果返回给客户端。只要在线服务器的数目多于两倍的故障服务器数目，系统可以通过算法1实现返回数据的正确一致性。

算法 1 数据一致性算法

步骤 1 获取全局时间戳

$$\text{tstart} = \text{TTCB\_getTimestamp}() + T_0$$

步骤 2 构造提议的消息

$$M = (\text{elist}, \text{tstart}, \text{data})$$

步骤 3 每个服务器提议一个消息，所有提议的消息中超过一半数目的相同消息是最终决定值，则有：

$$\text{propose} = \text{TTCB\_propose}(\text{elist}, \text{tstart}, \text{TTCB\_TBA\_MAJORITY}, H(M))$$

步骤 4 决定最终的提议值

$$\text{decide} = \text{TTCB\_decide}(\text{propose.tag})$$

步骤 5 如果服务器所提议消息的哈希值 $H(M)$ 与最终决定值相同，即 $H(M) = \text{decide.value}$，则服务器返回结果 data；否则，返回任意值。

参数elist表示所有参与服务的进程列表；tstart为通过时间服务获取的时间戳。在理想情况下，服务应该在elist中所有进程都提议一个值之后才能执行，但是恶意进程可能无限期地延迟提议时间，导致服务不能执行。为了避免这种情况发生，在tstart时间后服务立即执行，并不再接受任何其他提议。data表示服务器即将返回给客户端的结果；TTCB_TBA_MAJORITY表示选取超过一半数目的相同数据作为最终的返回结果；$H(M)$表示消息$M$的哈希值；tag是由TTCB_propose返回的唯一标识符，用于标识一个服务执行实例。

控制通道使得各个本地TTCB之间可以安全地传输各自的提议值，保证传输数据的完整性。算法1选取超过一半数目的相同数据作为最终的返回结果，为了满足数据的正确一致性，在最大故障服务器数目为 $f$ 的情况下，在线服务器中正确服务器的数目必须不小于 $2f+1$。TTCB在tstart后不再接受提议数据，并继续执行下一步协议，从而避免了恶意服务器无限地延迟提议数据，以达到拒绝服务的目的。

### 1.2 中央控制器

中央控制器主要是控制服务器的在线和离线状态。每隔一个时间间隔 $\Delta T$，中央控制器按照顺时针方向依次改变服务器的状态，如图2所示。令 $\Delta T$ 为一个单位时间，服务器上线速率 $v_{\text{on}} = t/\Delta T$，离线速率 $v_{\text{off}} = q/\Delta T$，入侵容忍系统的服务器总数为 $N$。

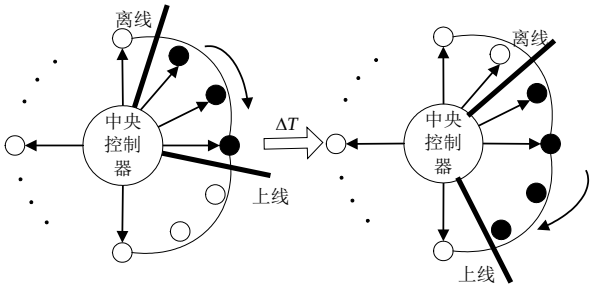



图2 服务器状态变化

由算法1可知，在 $\Delta T$ 内，在线服务器的数目必须满足 $N_{\text{on}} \geqslant 2f+1$，才能满足数据的正确一致性。$f$ 为在线服务器可以容忍的最大故障服务器数目，令初始状态时在线服务器的数目为 $N_{\text{on}}^0 = 2f+1$，$t$ 时刻在线服务器的数目为 $N_{\text{on}}^t$。为了增强系统容忍入侵的能力，中央控制器将动态调整在线服务器的数目。服务器的状态控制如算法2所示。

算法 2 服务器状态控制算法

步骤 1 设 $T_0$ 时刻为初始状态，在线服务器的数目 $N_{\text{on}}^0 = 2f+1$，$v_{\text{on}} - v_{\text{off}} \geqslant 1$；

步骤 2 顺时针方向以速率 $v_{\text{on}}$、$v_{\text{off}}$ 依次改变服务器的状态；

步骤 3　在 $T=T_0+\dfrac{N-(2f+1)-1}{v_{\text{on}}-v_{\text{off}}}\Delta T$ 时刻，将上线速率和离线速率互换，即 $v_{\text{on}}\leftarrow v_{\text{off}}$，$v_{\text{off}}\leftarrow v_{\text{on}}$；同时，$T_0\leftarrow T$，$N_{\text{on}}^0\leftarrow N_{\text{on}}^T$；

步骤 4　重复步骤2和3。

对于任意时刻 $t$，在线服务器的数目 $N_{\text{on}}^t=N_{\text{on}}^0+(v_{\text{on}}-v_{\text{off}})(t-T_0)$，其值变化范围为 $2f+1\sim N-1$，如图3所示。图中，二分之一个变化周期 $T_p=\dfrac{N-(2f+1)-1}{v_{\text{on}}-v_{\text{off}}}\Delta T$。

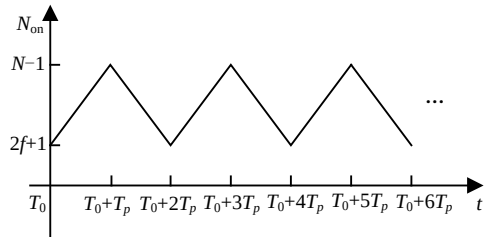



图3　在线服务器数目变化

攻击者必须在一个时间间隔内成功入侵超过一半数目的在线服务器，才能破坏数据的正确一致性。但是，中央控制器可以通过调整时间间隔以及上线和离线速率改变在线服务器的数目，使得攻击者很难预测时间间隔和在线服务器，增加了攻击的难度，提高了系统的安全性。系统以一定的复制策略[10]对离线的故障服务器进行恢复，因而可以提高系统的容忍能力和可用性。

## 2　仿真结果分析

本文主要从系统的可用性分析了入侵容忍系统的性能。假定在线服务器数目为 $2f+1$，将系统的可用性定义为：在一定时间间隔 $\Delta T$ 内，不出现大于 $f$ 个在线服务器同时发生故障所占的时间比率。本文利用Möbius工具[11-12]对系统的可用性进行仿真；分别分析了系统可用性与故障服务器的恶意行为概率 $P_{\text{mis}}$ 和攻击速率 $R_{\text{att}}$ (攻击者单位时间内攻击的次数)之间的关系。实验中，时间单位为分钟，时间间隔 $\Delta T=30$ min，实验结果的置信度为95%。

图4表明，该系统的可用性随着攻击速率的增加而逐渐降低，主要原因是攻击速率的增加，使得单位时间内发生故障的服务器数目增加，减少了可用的在线服务器数目，因而系统的可用性降低。在线服务器数目的增加可以提高该系统的可用性，因为在故障服务器数目相同的情况下，系统可用的服务器数目增加。

图5表明，系统对故障服务器的恶意行为非常敏感。一旦某台服务器表现出恶意行为，系统可以迅速检测，并通过中央控制器使其处于离线状态，相应地增加在线服务器，因而系统的可用性迅速提高。

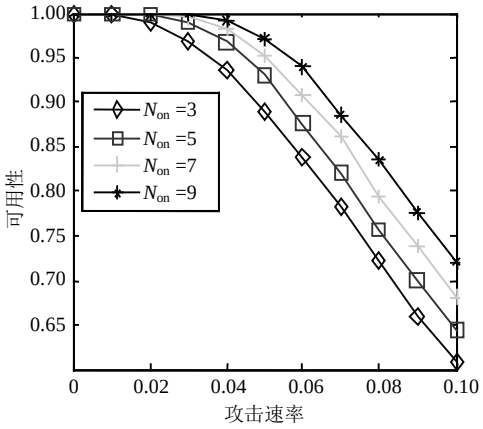



图4　攻击速率与可用性之间的关系

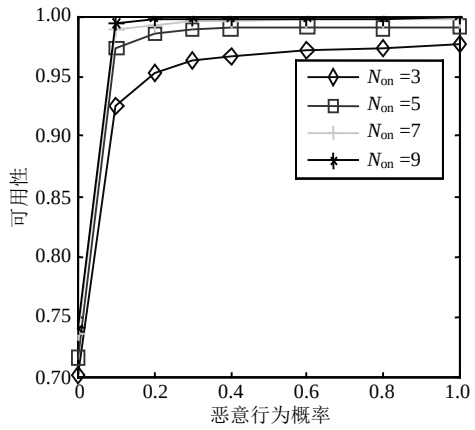
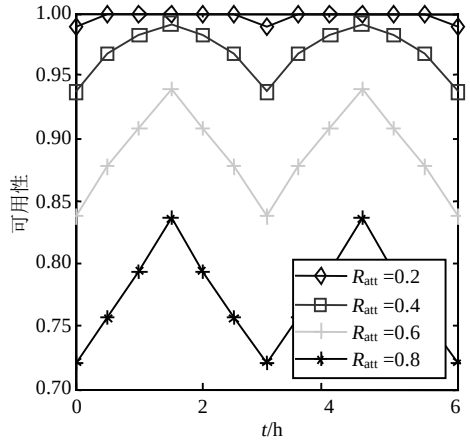



图5　恶意行为概率与可用性之间的关系

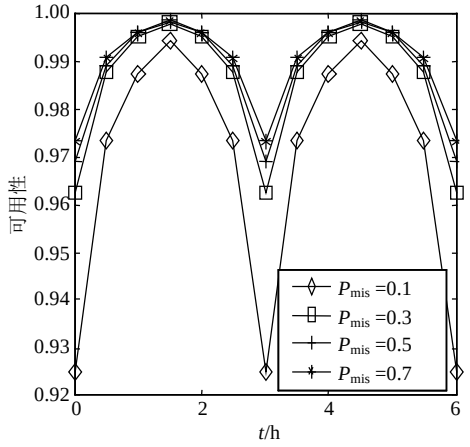
为了研究系统可用性随时间变化的关系，本文设定系统服务器的数目 $N=10$，初始状态时在线服务器可以容忍的故障服务器最大数目 $f=1$，上线速率为 $3/\Delta T$，离线速率为 $1/\Delta T$，因而 $T_p=1.5$ h。图6a和图6b所示分别为在不同攻击速率和恶意行为概率下，两个周期的系统可用性变化曲线。

图6a说明攻击速率越大，系统的可用性越低。图6b表明恶意行为概率越大，系统可用性越高。因为恶意行为概率的增加表明其被检测的概率增加，系统可以更迅速地采取入侵容忍措施屏蔽故障服务器的恶意行为，所以系统的可用性增加。系统的可用性在时间区间 $[1+2nT_p,2+2nT_p]$ 内较好，其中 $n=0,1,2,\cdots$。因此，可以根据实际需要将系统可用性最高的时间区间调整到系统服务最繁忙的时段，不仅可以增加系统的入侵容忍能力，并能提高对客

户端的服务能力。




a. 可用性变化与攻击速率的关系




b. 可用性变化与恶意行为概率的关系

图6 可用性变化曲线

## 3 结 论

入侵容忍作为信息安全领域前沿的研究课题，正越来越受到重视。本文提出了一种基于TTCB的环形入侵容忍系统，它具有以下优点：

(1) 当在线服务器数目大于两倍的故障服务器数目时，该系统实现了服务数据的正确一致性。

(2) 该系统可以动态地改变在线服务器的数目，增加了攻击难度，提高了系统的安全性。

(3) 离线故障服务器进行恢复时，该系统仍然可以对外提供服务，保证了系统的不间断服务能力。

本文利用仿真实验分别分析了系统可用性与攻击速率和恶意行为概率之间的关系，并得出了系统可用性较好的时间区间，为进一步提高系统的服务能力提供了一定的依据。

## 参 考 文 献

[1] 周 华, 孟相如, 杨茂繁, 等. 入侵容忍系统的状态转移模型定量分析[J]. 北京邮电大学学报, 2008, 31(3): 94-97.
ZHOU Hua, MENG Xiang-ru, YANG Mao-fan, et al. Quantifying the state transition model of intrusion tolerance system[J]. Journal of Beijing University of Posts and Telecommunications, 2008, 31(3): 94-97.

[2] CORREIA M, NEVES N F, LUNG L C. Byzantine-resistant consensus based on a novel approach to intrusion tolerance[C]//Fast Abstract in Supplement of the 10th Pacific Rim International Symposium on Dependable Computing. Tahiti, French: IEEE Press, 2004.

[3] MONIZ H, NEVES N F, CORREIA M. Randomized intrusion-tolerant asynchronous services[C]//DSN'06: Proceedings of the International Conference on Dependable Systems and Networks. Philadelphia, PA: ACM Press, 2006.

[4] CORREIA M, VERISSIMO P, NEVES N F. The design of a COTS real-time distributed security kernel (extended version) [R]. DI/FCUL TR-01-12. Portugal, 2001.

[5] VERISSIMO P, CASIMIRO A. The timely computing base model and architecture[J]. IEEE Transactions on Computers, 2002, 51(8): 916-930.

[6] CORREIA M, NEVES N F, LUNG L C, et al. Worm-IT—A wormhole-based intrusion-tolerant group communication system[J]. Journal of Systems and Software, 2007, 80(2): 178-197.

[7] CORREIA M, NEVES N F, VERISSIMO P. How to tolerate half less one byzantine nodes in practical distributed systems[C]//Proceedings of the 23rd IEEE Symposium on Reliable Distributed Systems. Florianpolis, Brazil: IEEE Press, 2004: 174-183.

[8] REITER M K. A secure group membership protocol[J]. IEEE Transactions on Software Engineering, 1996, 22(1): 31-42.

[9] LUNG L C, CORREIA M, NEVES N F. A simple intrusion-tolerant reliable multicast protocol using the TTCB[EB/OL]. [2007-09-11]. http://www.di.fc.ul.pt/~nuno/PAPERS/SBRC03.pdf.

[10] ZHOU Xu, LU Xian-liang, HOU Meng-shu, et al. Research on distributed dynamic replication management policy[J]. Journal of Electronic Science and Technology of China, 2005, 3(2): 97-102.

[11] SANDER W H. Möbius[CP/OL]. [2007-06-10]. http://www.mobius.uiuc.edu.

[12] GUPTA V, LAM V, RAMASAMY H V, et al. Dependability and performance evaluation of intrusion-tolerant server architectures[C]//Proceedings of the 1st Latin-American Symposium on Dependable Computing. Sao Paulo, Brazil: Springer, 2003: 81-101.

编 辑 黄 莘