文章编号:1671-8860(2007)09-0842-04　　**文献标志码**:A

基于正常轮廓更新的自适应异常检测方法

熊　平[1]

(1　中南财经政法大学信息学院,武汉市武珞路 114 号,430060)

摘　要:根据网络系统发生正常改变的基本特征,提出了确定网络系统正常改变的"三条件"计算方法,其计算结果可作为更新正常轮廓的依据。对正常轮廓的更新问题进行了深入探讨,提出了自适应异常检测的实现机制。并以网络流量分析为例,验证了在异常检测中应用这一方法的可行性。

关键词:异常检测;正常轮廓;规则更新

中图法分类号:TN91

入侵检测系统(IDS)是信息安全体系的重要组成部分。入侵检测的方法主要分为两种,即滥用检测和异常检测[1]。其中,异常检测由于能够检测到未知及新型攻击,是一种更为严格的入侵检测方法。目前,异常检测存在的主要问题是误报率较高。导致这一问题的主要原因是用于异常检测的正常轮廓没有得到及时的更新。针对这一问题,文献[2]提出了一个应用数据挖掘来实现自适应入侵检测的方法;文献[3]对系统正常改变的识别进行了初步研究[3]。这些框架性方案为自适应异常检测方法的深入研究建立了基础。本文根据系统正常改变的特征,提出了用于确定系统正常改变的"三条件"计算方法。

1　异常检测及其自适应问题

异常检测是基于这样一个假设,即入侵者的活动在某种程度上与正常用户的行为有所不同[4]。因此,检测异常和入侵实质上就是检测这些特性参数与正常状态值的背离程度[5]。用相似度来表示系统当前状态与正常轮廓的背离程度,以确定当前系统是否处于异常状态[6]。若相似度大于预先定义的阈值,表示网络在该时间窗内处于正常状态,否则处于异常状态。可以看出,判断当前系统是否处于异常状态是以系统的正常轮廓为基本参考的。一般来说,系统的正常变化是缓慢的、细微的、长期的,称之为"渐变",因此,用于异常检测的系统正常状态的"轮廓"即正常状态规则集也必须随之不断地作出相应的更新,以真实地表征系统的当前正常状态,这就是异常检测系统的自适应问题。

2　识别系统的正常改变

2.1　定义系统的正常改变

在异常检测过程中,检测系统会连续地对时间窗(可连续或交迭)内的网络数据进行处理,相应地得到每个时间窗相应规则集与正常轮廓的相似度,将这些相似度与时间窗一一对应,就形成了一条相似度-时间折线,在宏观上表现为一条平滑的曲线,如图 1 所示。其中,每个时间窗 ΔT 又包含多个时间子窗 Δt,统计每个时间子窗 Δt 里系统的状态属性,就构成了一条系统状态记录,全部记录则组成了系统在时间窗 ΔT 里的行为数据

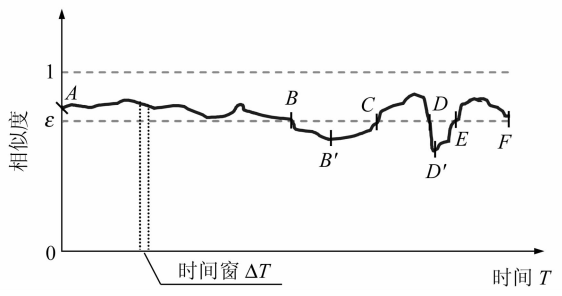

图 1　相似度随时间的变化

Fig. 1　Similarity Changes over Time

收稿日期:2007-06-25。

项目来源:中南财经政法大学引进人才科研启动基金资助项目。

库。挖掘这个网络行为数据库，得到规则集，进而可得到时间窗 ΔT 内系统状态与正常轮廓的相似度。将每个时间窗 ΔT 的相似度连成图 1 中的曲线，即可反映出相似度随时间的变化情况。

在图 1 中，线段 BC 和 DE 所在的每个时间窗的相似度都低于阈值 ε，因此，异常检测系统将在线段 BC 和 DE 所在的每个时间窗报警，并根据相似度大小采取相应的响应。

然而，相似度小于阈值的原因有两种可能，即系统状态发生了正常改变，或者存在网络入侵。由于系统状态的正常改变过程通常是缓慢的、细微的，因此，完全可以用相似度曲线在单位时间内的变化率来表示系统状态的变化程度。如果变化率很大，曲线表现得很陡峭，则认为是入侵导致了系统状态的变化，检测系统将根据预先定义的规则进行响应；反之，如果变化率很小，曲线表现很平缓，则认为是系统状态的正常变化，这时，检测系统不应作出响应，而应该调整网络的正常轮廓。如在图 1 中，线段 BB' 的变化很平缓，可以确定为正常的改变，而线段 DD' 变化很陡峭，可认为是网络入侵导致的结果，无需更新正常轮廓。

2.2 “三条件”计算方法

为了使异常检测系统能够自动地识别系统状态的正常改变，必须根据以上提出的系统状态正常改变的原理建立适当的数学模型，并形成相应的程序模块固化到检测系统中去，使系统能够按照确定的数学计算方法来精确地识别正常的改变。

设相似度折线上两个相邻的点为 $A(T_1, s_1)$、$B(T_2, s_2)$，其中，s_1 和 s_2 分别是时间为 T_1 和 T_2 时的相似度，$T_2 - T_1 = \Delta T$ 为时间窗的长度。则两点连线的斜率 $k = (s_2 - s_1)/\Delta T$ 表示相似度在时间窗内的变化率。设两个相邻时间窗内相似度的变化率分别为 k_1 和 k_2，则 $a = (k_2 - k_1)/2\Delta T$ 表示相似度在这两个相邻时间窗内的变化程度。

考察系统状态正常改变时相似度折线所具有的特征，可以归纳为以下两点：① 在每个时间窗内，相似度线段不能下降得过于陡峭。如果在某个时间窗内，相似度线段下降得过于陡峭，则定义为入侵，即后续时间窗的相似度不会比前一个时间窗的相似度小很多。这可以用时间窗首尾两点的斜率 $k \leqslant \delta$ 来表示，δ 是一个预定义的值。② 若干个连续点连接而成的折线在最小相似度阈值线之下，但与最小相似度阈值线之间的距离不大。显然，必须通过对多个连续时间窗的相似度进行跟踪分析，才能够判断相似度折线是否满足这一特性。而参数 a 仅能表示相邻两个时间窗的变化，因此，必须采用其他方式确定一个反映这一特性的参数。这里提出了一个用面积来度量这一特性的方法。

考察一条相似度折线中连续多个时间窗的相似度都小于阈值 ε 的部分，如图 2 中的折线 AC-$DEFGHI$。相似度在时间窗 AC 处由大于阈值 ε 转为小于阈值 ε，线段 AC 与阈值 ε 水平线的交点为 B。考察其在 x 个连续时间窗内的折线，如 6 个时间窗的折线 $ACDEFGH$，计算多边形 BC-$DEFGH\,H''$ 的面积为 S（即图 2 中阴影部分的面积）。显然，如果折线 $ACDEFGH$ 越陡峭，如折线 $A'C'D'E'F'G'H$，则其相应的面积（即多边形 $B'C'D'E'F'G'H'H''$ 的面积）就越大。因此，用相似度折线在阈值 ε 水平线下的面积来作为折线陡峭度的数学度量。

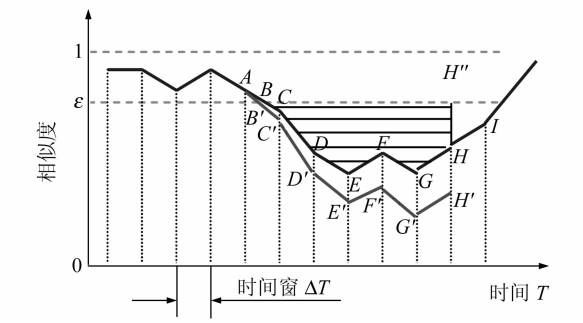

图 2　相似度曲线的下降程度分析

Fig. 2　Degression Degree of Similarity Curves

基于以上分析，这里提出了用于确定系统状态正常改变的“三条件”计算方法（three-premise computing approach，TPCA）。

当且仅当相似度折线满足以下三个条件时，则确定系统状态发生了正常改变，需更新用于异常检测的正常轮廓。

1）在相似度折线上存在连续的 $\alpha+1$ 个点 $X_0(T_0 S_0)$、$X_1(T_1, S_1)$、…、$X_\alpha(T_\alpha, S_\alpha)$，表示在一个由 $\alpha+1$ 个连续的时间窗 ΔT_0、ΔT_1、…、ΔT_α 组成的时间段 T 中，各时间窗所对应的相似度为 s_0、s_1、…、s_α，其中，s_0 大于相似度阈值 ε，s_1、…、s_α 均小于 ε。

2）将 $\alpha+1$ 个点按顺序点连接起来，组成一条由 α 条有向线段 $\overrightarrow{X_0X_1}$、$\overrightarrow{X_1X_2}$、…、$\overrightarrow{X_{\alpha-1}X_\alpha}$ 组成的折线。每一条线段的斜率 $k_{\overrightarrow{X_0X_1}}$、$k_{\overrightarrow{X_1X_2}}$、…、$k_{\overrightarrow{X_{\alpha-1}X_\alpha}}$ 均不能小于预定义的最小斜率 $k(k<0)$。

3）取有向线段 $\overrightarrow{X_0X_1}$ 与相似度阈值水平线 $s=\varepsilon$ 的交点为 $X'_0(T'_0, \varepsilon)$，将折线的终点 $X_\alpha(T_\alpha, s_\alpha)$ 映射到相似度阈值水平线 $s=\varepsilon$ 上，得到映射点 $X'_\alpha(T_\alpha, \varepsilon)$，最后求多边形 $X'_0X_1X_2\cdots X_\alpha X'_\alpha$ 的面积

S_T，S_T应不大于预定义的最大面积Φ，即 $S_T \leqslant \Phi$。

显然，TPCA 是通过三个阈值 α、k、Φ 来确定系统正常改变的。这些参数的取值需要在系统的实际运行中通过大量的统计试验来确定，并可通过遗传算法再优化。

3 正常轮廓的更新

在识别到系统状态发生了正常改变之后，必须对正常轮廓进行更新，以保证异常检测的准确性。正常轮廓是通过对系统正常状态训练数据集进行挖掘得到的，因此，更新正常轮廓归根结底在于更新训练数据集，对调整后的训练数据集进行数据挖掘，即可得到新的关联规则集，即新的正常轮廓。

当确定网络环境发生了正常改变时，检测系统收集发生在正常改变的时间窗里网络环境的各种属性，形成新的记录，打上时间戳并添加到训练数据集中，同时从训练数据集中删除那些太“旧”而过时的记录。这样就实现了训练数据集的更新，然后通过数据挖掘，得到更新的正常轮廓。

另外，由于正常轮廓的更新发生在实时检测中，为了不影响检测效率，必须考虑其计算代价和处理速度。在异常检测系统中，可应用高效的关联规则挖掘算法[7]来实现正常轮廓的快速更新。

4 试 验

运用以上阐述的方法，对试验环境下局域网的流量进行分析。首先，选择与网络流量相关的 4 个属性来对系统进行分析，即 TCP 包和 UDP 包在全部数据包中的比例 P_{tcp} 和 P_{udp}、网络中每 s 的平均数据包数量 Avg. packet/s 以及每 s 的平均数据位 Avg. Mbit/s。在时间段 9:00 至 12:00 中，每 2 min 统计一次这些网络属性的值，得到包含 90 个事务的训练数据集 D。对训练数据集 D 作数据挖掘，产生关联规则集 S_0，作为网络流量的正常轮廓。然后，应用这一正常轮廓对 9:00 至 21:00 的网络流量进行实时检测，得到多条相似度曲线(如图 3)。

由图 3 可以看出，网络流量在 12:00 之后逐渐与正常轮廓发生偏离，根据曲线的数学特征，最终归纳出网络流量发生正常改变的三个条件为：① 在相似度折线上存在连续的 7 个点 $X_0(T_0, s_0)$、$X_1(T_1, s_1)$、…、$X_6(T_6, s_6)$，其中，s_0 大于相似度阈值 0.6，s_1、…、s_6 均小于 0.6；② 由该 7 个点

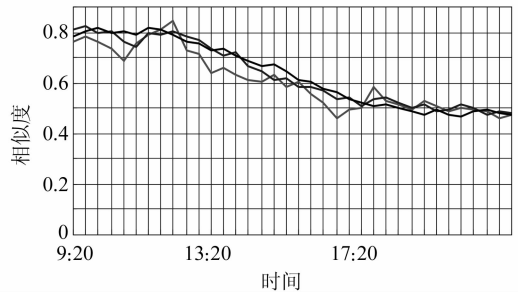

图 3 网络流量检测的相似度曲线

Fig. 3 Similarity Curves of Network Traffic Detection

组成的折线段上，各组成线段的斜率均不能小于预定义的最小斜率 −0.2；③ 取有向线段 $\overrightarrow{X_0X_1}$ 与相似度阈值水平线 $s=\varepsilon$ 的交点为 $X'_0(T'_0, 0.6)$，将折线的终点 $X_6(T_6, s_6)$ 映射到相似度阈值水平线 $s=0.6$ 上，得到映射点 $X'_6(T_6, 0.6)$，则多边形 $X'_0X_1X_2\cdots X_6X'_6$ 的面积 $S_T \leqslant 10.5$。

将以上得到的确切条件应用到实际检测中。按照前面的方法，收集网络数据(其间在某时段对一台主机作拒绝服务攻击)，并计算各时段的相似度，得到相似度折线如图 4 所示。

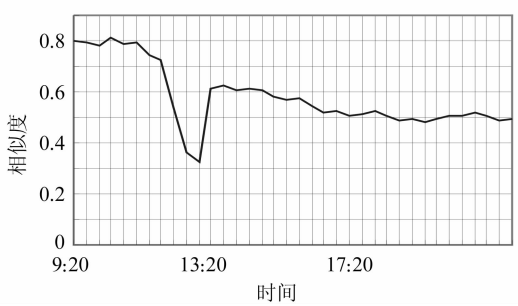

图 4 实时检测中的相似度曲线

Fig. 4 Similarity Curve in Real-Time Detection

根据前面确定的判定系统状态正常改变的三个条件，对图 4 中的相似度折线进行分析。从图 4 可以看出，从 12:00 至 13:00 的拒绝服务攻击使相似度折线降低到了阈值水平线 $s=0.6$ 之下，如图 5 所示。该折线段的前两条线段的斜率分别

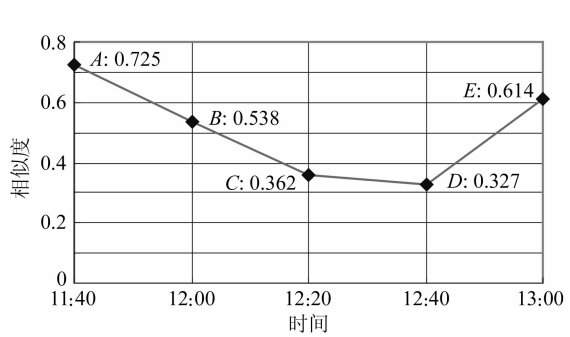

图 5 拒绝服务攻击下的相似度折线

Fig. 5 Similarity Curve with DoS Attack

为 $k_{\overrightarrow{AB}}=-0.561$、$k_{\overrightarrow{BC}}=-0.528$,均小于预定义的最小斜率−0.2,因此不符合正常改变的条件,不会被系统判定为系统状态的正常改变。

而对于图 4 中的相似度折线 14:20 之后在阈值水平线 $s=0.6$ 下的部分,考察其连续 7 个点的折线段,如图 6 所示。计算该折线段上各组成线段的斜率分别为 $k_{\overrightarrow{FG}}=-0.087$、$k_{\overrightarrow{GH}}=-0.039$、$k_{\overrightarrow{HI}}=-0.018$、$k_{\overrightarrow{IJ}}=-0.084$、$k_{\overrightarrow{JK}}=-0.078$、$k_{\overrightarrow{KL}}=-0.018$。全部线段的斜率均大于预定义的最小斜率−0.2,即满足了条件②。

计算多边形 $F'GHIJKLL'$ 的面积 $S_T=0.152+0.55+0.62+0.84+1.38+1.58=5.122$。显然,$S_T$ 小于预定义的最大面积 $\Phi=10.5$,即满足条件③。由此可确定,网络流量在 14:20 之后开始发生了正常变化。

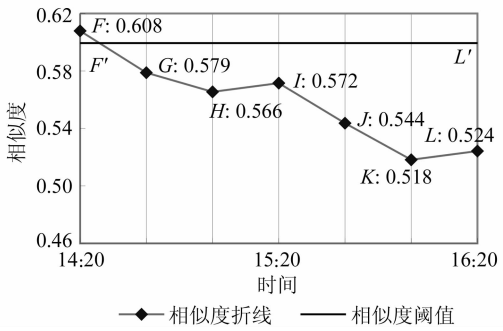

图 6　正常改变时的相似度折线

Fig. 6　Similarity Curve with Normal Change

由以上结果可以看出,根据系统状态正常变化的三个判定条件,能够有效地将入侵导致的系统状态变化与系统状态的正常变化区分开来,为更新系统状态的正常轮廓提供正确的依据。

参　考　文　献

[1] Stefan A. Intrusion Detection Systems: A Survey and Taxonomy[R]. Technical Report No 99-15, Dept. of Computer Engineering, Chalmers University of Technology, Sweden, 2000

[2] Wenke L, Salvatore J S. Adaptive Intrusion Detection: a Data Mining Approach[J]. Artificial, Intelligence Review, 2001, 14:533-567

[3] Hossain M, Bridges S M. A Framework for an Adaptive Intrusion Detection System with Data Mining[C]. The 13th Canadian Information Technology Security Symposium (CITSS 2001), Ottawa, Canada, 2001

[4] Taghi M K, Mohamed E A. Resource-Sensitive Intrusion Detection Models for Network Traffic[C]. The 8th IEEE International Symposium on High Assurance Systems Engineering, Tampa, Florida, 2004

[5] Li Kunlun, Huang Houkuan, Tian Shengfeng, et al. Fuzzy Multi-class Support Vector Machine and Application in Intrusion Detection[J]. Journal of Computers, 2005, 28(2): 274-280(in Chinese)

[6] Estevez-Tapiador J M, Garcia-Teodoro P, Diaz-Verdejo J E. Anomaly Detection Methods in Wired Networks: a Survey and Taxonomy[J]. Computer Communications, 2004, 27(16): 1 569-1 584

[7] Song Mingjun, Sanguthevar R. Finding Frequent Itemsets by Transaction Mapping[C]. The 2005 ACM Symposium on Applied Computing, Santa Fe, New Mexico, 2005

作者简介:熊平,博士。主要研究方向为通信系统与网络安全。
E-mail:pingxiong01@126.com

An Adaptive Anomaly Detection Method Based on Normal Profile Updating

XIONG Ping[1]

(1　School of Information, Zhongnan University of Economics and Law, 114 Wuluo Road, Wuhan 430060, China)

Abstract: According to the characteristics of legal change in Network, a three-premise computing approach is brought out to identify the legal change of protected Network. The problem of normal profile updating is discussed and the principle that designing an adaptive anomaly detection system is described. Using experiments on network traffic analysis, the feasibility of updating normal profile for anomaly detection system is validated.

Key words: anomaly detection; normal profile; rules updating

About the author: XIONG Ping, Ph.D. His major research orientation is information network security.
E-mail: pingxiong01@126.com