一种基于个体经验的多粒度信任模型

邓忠军[1] 王少杰[2] 郑雪峰[1] 锁延峰[1,2] 于 真[1]
(北京科技大学信息工程学院 北京 100083)[1] (国家信息技术安全研究中心 北京 100094)[2]

摘 要 分布式网络中,对于某一节点所提供的相同质量的服务,不同的访问节点对该节点的信任评价存在差异。导致这种差异的原因,一方面与访问节点的直接交互经验有关,另一方面与访问节点的兴趣爱好及对服务评价的理解角度有关(有的节点对服务的评价看重的是下载速度,而有的节点则更看重服务的安全可靠等),这种差异必然影响信任评价的准确性。为了消除个体节点信任评价差异所产生的影响,通过引入经验因子的方法和采用多元组的信任信息记录方法,提出了一种基于个体经验的多粒度信任模型。实验分析表明,该模型在信任评价的粒度、信任评价的准确性等方面有较大的提高。

关键词 信任模型,非对称性,相对经验因子,反馈可信度
中图法分类号 TP393 **文献标识码** A

Multi-granularity Trust Model Based on Individual Expericence

DENG Zhong-jun[1] WANG Shao-jie[2] ZHENG Xue-feng[1] SUO Yan-feng[1,2] YU Zhen[1]
(School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China)[1]
(National Research Center for Information Technology Security, Beijing 100094, China)[2]

Abstract In distributed networks, the trust evaluation for the same quality services of one node conducted by different access nodes is different. On the one side, it has relation to the direct mutual experience of the accessing node. And on the other side, it has relation to the accessing node's interest and opinion about the service evaluation. Some nodes think a lot of the download speed, while some other nodes put more emphasis on the security and reliability of the service, which cannot but lead to influence the veracity of trust evaluation. In order to eliminate the infection resulted from the trust evaluation difference of individual nodes, a multi-granularity trust model based on individual experience was put forward. In this model, experience factor and multi trust information recording methods were employed. Experiment and analysis show that the proposed model has better improvement in granularity and veracity of trust evaluation.

Keywords Trust model, Asymmetry, Relative experience factor, Feedback trust value

1 引言

分布式网络中存在这样的问题,访问节点 i 询问具有相同经验的节点 k_1 和 k_2,服务节点 j 所提供的服务如何,假设节点 i 最看重服务的质量。节点 k_1 根据自身的经验,善意真实地告诉节点 i:服务节点 j 提供服务的能力为75%,假设节点 k_1 最看重服务的下载速度;同时节点 k_2 根据自身的经验,也很善意并且真实地告诉节点 i:服务节点 j 提供服务的能力为90%,假设节点 k_2 最看重的是服务的可靠性。在这种情况下,节点 i 应该如何处理真实的评价值却不相同的反馈信息?从理论上而言,虽然节点 k_1 和 k_2 都提供着真实的信任评价,但节点 i 容易认为节点 k_1 和 k_2 所提供的是恶意评价,这是因为节点 i 所询问的主要是服务质量如何,而节点 k_1 反馈的更主要的是对 j 服务下载速度的满意度评价(75%),节点 k_2 则更多的是对 j 服务可靠性的满意度评价(90%),在这样答非所问的情况下,一方面会误导访问节点 i 对服务节点 j 提供真实服务的评价,同时也会导致提供真实反馈的节点反馈可信度的降低。

这些问题是访问节点评价标准差异所引起的。设定某一个特定的服务节点,在特定的时间内提供了稳定的服务,假设在此期间该服务节点提供服务的安全性(病毒及恶意插件、数据的保密性…)、服务质量(下载速度、服务的响应时间…)、可靠性(建立连接的成功率、无故障服务的次数…)等方面均保持一致。但对于不同服务的请求者,对该服务的满意度进行信任评价时可能产生不同的评价值,导致这种差异的原因可以分为以下几类:

(1) 访问节点的经验不同,导致服务评价的差异;

(2) 服务请求者对服务满意度的预期值的要求不一致,

到稿日期:2009-06-12 返修日期:2009-08-24 本文受国家"八六三"高技术研究发展计划项目基金(No. 2007AA012474),国家发改委信息安全专项基金(No. [2008]1736)资助。

邓忠军(1963-),男,博士生,主要研究方向为网络安全、信任算法、数据挖掘等,E-mail:haoyizz@163.com;王少杰(1976-),男,博士,主要研究方向为风险评估、等级保护、入侵检测等;郑雪峰(1951-),男,博士生导师,主要研究方向为信息安全、网络安全;锁延峰(1977-),男,博士生,主要研究方向为信任、风险评估、入侵检测;于 真(1983-),女,博士生,主要研究方向为网络安全、P2P。

以下载速度为例，有的服务请求者认为下载速度达到 20kpbs 时，对下载速度的满意度就可以达到 100%，而有的节点认为下载速度达到 20kpbs 时，对下载速度的满意度就只能达到 20%，只有当下载速度达到 1Mbps 或以上时，对下载速度的满意度才为 100%。

(3) 服务请求者对服务满意度评价标准的不一致，例如有的访问者最看重服务的响应时间，然后是下载速度、无故障服务的时间等因素，而有的访问者则最看重的是服务有无病毒及恶意插件，然后是无故障服务的时间、服务响应时间、下载速度等因素。访问节点对服务评价的各个因素看重的情况不一致，会导致在服务节点提供的下载速度、无故障服务的时间、响应时间等因素完全相同的情况下，不同节点对该服务的评价仍然存在差异。

为了解决上述问题，提出了一种新的基于个体经验的多粒度信任模型。仿真实验分析表明，新模型能够体现个体评价的差异，提高信任评价的准确度。

2 相关工作

众学者对信任模型进行了不同角度、不同方面的研究工作，例如 Beth 等先提出了信任定量化的概念和方法[1,2]，将信任分为直接信任和推荐信任，根据肯定和否定经验数计算实体完成任务的概率，以此表示信任，并给出了信任合成的方法。Rahman 等人提出的信任度评估模型同样将信任关系分为直接信任和推荐信任，给出了信任的传递协议和计算公式，但没有给出信任综合计算公式。Jøsang 等提出主观逻辑的方法，其实质是利用了证据理论(D-S 理论)[3,4]。Mui 等从社会学和进化论的角度给出了一个信任和信誉的计算模型[5]。同时，其他一些学者也采用不同的方法来度量和推理信任关系，比如模糊理论[6]等等。

在基于反馈信息的研究方面，信任模型大致可分为全局信任模型和局部信任模型。全局信任模型又可分为两类：一类[7]是根据节点获得的正面反馈和负面反馈的数目，进行简单的算术运算，得出节点的全局可信度。该方法计算简单易理解，但无法处理节点给出的不正确评价，容易受到恶意节点的联合欺诈攻击；另一类[8,9]是通过对信任传递链上的信任值重复迭代来计算网络中节点的信任值。这种方法需要节点之间合作处理信任信息，计算和通信开销都较大。全局信任模型忽略了信任的私人化特征，对于某个特定的节点，其他节点对它的信任值都是相同的。此外，在大规模的 P2P 网络中为每个节点计算全局信任值的必要性和可行性仍有待进一步研究[10]。

在基于共享信息的信任研究方面，多数信任机制[11-14]的原理来源于社会关系网络[15]，因此分布式网络中节点间的信任关系与社会网络中人与人之间的信任关系有很大相似性，现有的关于 P2P 网络的信任模型大多属于此类[16,17]。该类信任模型共享信息的获取有两种途径：一种是通过向其他节点洪泛信任请求获得的，该方法可扩展性差；另一种是通过采用 DHT 机制的 P2P 存储系统如 Chord[18]，P-Grid[19]等获得的，这种方法不适合节点频繁加入和离开系统的 P2P 系统。

现有的信任模型不能解决访问节点评价标准差异所引起的问题。

3 信任模型

为了方便讨论，对于一个节点所提供的服务进行信任评价时，这里只考虑病毒及恶意插件、平均数据包下载速度、平均服务的响应时间、平均无故障响应时间、提供服务的时间段 5 个因素。

定义 1 六元组 $S^x=(X^0,X^A,X^B,X^C,X^D,X^E)$ 是描述节点 X 的服务性能的集合，节点 X 的服务的描述可以用表的形式表示，如表 1 所列。

表 1 描述节点 X 的服务性能的集合

服务种类	病毒及恶意插件	数据包下载速度	服务响应时间	无故障响应时间	提供服务的时间段
X^0	X^A	X^B	X^C	X^D	X^E

其中，X^0 表示服务的种类，表示节点 X 能提供哪种类型的服务。

X^A 的取值为 0 或者 1，取值为 0 时表示有病毒或恶意插件，取值为 1 时表示无病毒和恶意插件；X^A 的取值为 0 时，表明服务是不安全的，取 1 时，服务是安全的。

X^B 的取值为某个正实数，单位是 bps；这里是指服务所提供资源的下载速度，X^B 的取值越大，服务的质量越高。

X^C 的取值为某个正实数，单位是 ms；X^C 的值指的是访问服务时所需要等待的时间，等待时间也就是服务器的响应时间，X^C 的数值越小，服务质量就越高。

X^D 的取值为一段时间间隔，指的是服务相邻两次故障之间的平均工作时间，也叫平均无故障间隔，X^D 的数值越大，服务的可靠性越高。

X^E 的取值为某个时间段，X^E 的值表示节点在某个时间段提供服务，在其他的时间段不提供服务。

假设节点 j 能够提供某个特定的服务 j^0，同时 j 发布的服务声明集为 $S^j=(j^0,j^A,j^B,j^C,j^D,j^E)$，节点 i 根据节点 j 的声明，对 j 提供的服务进行访问，直接交互后得到的结果表示为六元组 $d_{ij}=(j^0,d_{ij}^A,d_{ij}^B,d_{ij}^C,d_{ij}^D,d_{ij}^E)$。

其中，d_{ij} 表示节点 i 对节点 j 的直接交互结果；

j^0 表示 j 所提供服务的种类，能够提供怎样的服务，同一个节点可能提供多类服务；

d_{ij}^A 表示节点 j 提供的服务中有无病毒和恶意插件，取值为 0 表示有，1 表示没有；

d_{ij}^B 表示节点 i 下载节点 j 所提供资源的实际下载速度；

d_{ij}^C 表示节点 i 等待节点 j 所提供服务的实际等待时间；

d_{ij}^D 表示节点 j 对节点 i 提供服务时的实际无故障时间；

d_{ij}^E 表示节点 i 访问 j 的时间段。

节点 i 根据直接交互结果 d_{ij}，对节点 j 所提供的服务进行信任评价，评价值用六元组 $D_{ij}=(j^0,D_{ij}^A,D_{ij}^B,D_{ij}^C,D_{ij}^D,D_{ij}^E)$ 表示，其中 $D_{ij}^A,B_{ij}^D,D_{ij}^C,D_{ij}^D,D_{ij}^E$ 为节点 i 根据自身对服务的要求及对信任评价的理解，对 j 提供服务的这 5 个方面的直接交互满意度评价。

节点 i 得到节点 j 的信任评价值 $D_{ij}=(j^0,D_{ij}^A,D_{ij}^B,D_{ij}^C,D_{ij}^D,D_{ij}^E)$ 后，根据自身对服务的需求和理解，同时结合自己的经验，对节点 j 的服务进行直接信任整体评价，评价值 WD_{ij} 表示为：

$$WD_{ij}=(D_{ij}^A D_{ij}^B D_{ij}^C D_{ij}^D D_{ij}^E)(w_A w_B w_C w_D w_E)^T \quad (1)$$

其中，w_1,w_2,w_3,w_4,w_5 分别表示节点 i 对 $D_{ij}^A,D_{ij}^B,D_{ij}^C,D_{ij}^D,$

D_{ij}^{E}采用的权重，满足$0\leqslant w_A,w_B,w_C,w_D,w_E\leqslant 1$，同时满足$w_A+w_B+w_C+w_D+w_E=1$。其中$w_A,w_B,w_C,w_D,w_E$的值可以通过如下的方法获取：

1）用户可以根据自己对服务的需求进行指定，用户更看重的因素取较大的权重值；

2）参考领域专家根据经验设置的通用权重系数；

3）借鉴 Saaty 等人[20]提出的成对比较的层次分析方法，把相关因素两两相互对比，对比时采用相对尺度，尽可能地减少性质不同的诸因素相互比较的困难，提高准确度。

这样通过式(1)，节点 i 可以得到提供 j^0 类服务的节点 j 的直接信任整体评价，对于提供 j^0 类服务的节点 j_1,j_2,j_3,j_4,j_5……节点 i 与其进行直接交互后，也可以得到对这些节点的直接信任整体评价，表示为 $WD_{ij1},WD_{ij2},WD_{ij3},WD_{ij4},WD_{ij5}$……节点 i 可以根据这些服务节点的直接信任整体评价，决定下次跟哪个节点合作，同时也可以考虑其他节点对服务评价的推荐信任信息，对服务节点进行综合信任评价。

每次交互后，节点 i 向存放信任评价信息的档案点[21,22]提交对 j 信任评价值 D_{ij}，本文采用提交的 $D_{ij}=(j^0,D_{ij}^{A},D_{ij}^{B},D_{ij}^{C},D_{ij}^{D},D_{ij}^{E})$而不是 WD_{ij} 的原因，是节点 i 在进行整体信任评价获取 WD_{ij} 值时，考虑了自身的主观因素，别的节点不容易理解这样的评价值。信任评价信息的存放如表 2 所列。

表 2　访问节点对 j 的直接信任评价表

k_x 对 j 的信任评价	$j^0,D_{k_xj}{}^{A},D_{k_xj}{}^{B},D_{k_xj}{}^{C},D_{k_xj}{}^{D},D_{k_xj}{}^{E}$
k_1 对 j 的信任评价	$j^0,D_{k_1j}{}^{A},D_{k_1j}{}^{B},D_{k_1j}{}^{C},D_{k_1j}{}^{E},D_{k_1j}{}^{E}$
k_2 对 j 的信任评价	$j^0,D_{k_2j}{}^{A},D_{k_2j}{}^{B},D_{k_2j}{}^{C},D_{k_2j}{}^{D},D_{k_2j}{}^{E}$
k_3 对 j 的信任评价	$j^0,D_{k_3j}{}^{A},D_{k_3j}{}^{B},D_{k_3j}{}^{C},D_{k_3j}{}^{D},D_{k_3j}{}^{E}$
k_4 对 j 的信任评价	$j^0,D_{k_4j}{}^{A},D_{k_4j}{}^{B},D_{k_4j}{}^{C},D_{k_4j}{}^{D},D_{k_4j}{}^{E}$
……	……
k_n 对 j 的信任评价	$j^0,D_{k_nj}{}^{A},D_{k_nj}{}^{B},D_{k_nj}{}^{C},D_{k_nj}{}^{D},D_{k_nj}{}^{E}$

根据表 2，对节点 j 所提供的服务有病毒或恶意插件的信任评价 T^{jA} 可用下面的公式进行验证，这里采用求和平均的方法判断 j^A 的可信值：

$$T^{jA}=\sum_{x=1}^{n}\frac{D_{kxj}{}^{A}}{n} \tag{2}$$

为了评价节点 j 提供服务资源的下载速度信任值，采用了如下方法：

$$T^{jB}=\sum_{x=1}^{m}\frac{C_{kxj}{}^{B}\times D_{kxj}{}^{B}}{\sum_{x=1}^{m}C_{kxj}{}^{B}} \tag{3}$$

式中，$C_{kxj}{}^{B}$ 表示节点 k_x 对节点 j 下载速度的信任评价的可信度，是采用节点 k_x 给出的信任评价值的权重，定义为：

$$C_{kxj}{}^{B}=\sum_{u=1}{}^{m}\frac{\varphi_{kukx}^{j}{}^{B}C_{ku,kx}^{j}{}^{B}}{\sum_{u=1}{}^{m}\varphi_{kukx}^{j}{}^{B}} \tag{4}$$

式(4)满足 $ku\neq kx$。$\varphi_{kukx}^{j}{}^{B}$ 表示相对经验因子 $\varphi_{kukx}^{j}{}^{B}=2^{-\left|\frac{m_o}{m_{kuj}}-\frac{m_o}{m_{kxj}}\right|}$，$m_{kuj},m_{kxj}$ 分别表示节点 ku,kx 与服务节点 j 直接交互的次数；m_o 表示节点是否有经验的门限值，当节点的经验大于 m_o 时，表示该节点有经验，否则表示该节点经验不足；$C_{ku,kx}{}^{B}$ 表示节点 k_u 对 k_x 推荐 j 服务的个体反馈可信度评价，表示为：

$$C_{ku,kx}{}^{B}=\begin{cases}C_{ku,kx}{}^{B}+\alpha\cdot\varphi_{ku,kx}^{j}{}^{B}\cdot 2^{-2\Delta D^{B}/\theta}, & d_{ku,kx}^{j}{}^{B}<\theta\\ C_{ku,kx}{}^{B}-\beta\cdot\varphi_{ku,kx}^{j}{}^{B}\cdot 2^{-(1-\Delta D^{B}/20)}, & \text{其他}\end{cases} \tag{5}$$

式(5)满足 $ku\neq kx$。其中 $C_{ku,kx}{}^{B}$ 的取值范围是[0,1]，当 $C_{ku,kx}{}^{B}<0$ 时，取值为 0；当 $C_{ku,kx}{}^{B}>1$ 时，取值为 1。α,β 分别为反馈可信度增加、减少因子的值，满足 $0<\alpha<\beta<1$，一个节点提供善意推荐时，反馈可信度值的增加比较缓慢，若提供恶意推荐，可信度值将快速降低。θ 为访问节点对推荐节点能容忍的最大评价偏差；$d_{ku,kx}{}^{B}$ 表示为 $d_{ku,kx}{}^{B}=2^{(m_o/m_{kuj}-m_o/m_{kxj})}\cdot\Delta D^{B}$，$\Delta DB$ 表示节点 k_u 和 k_x 对 j 的下载速度的评价差异：

$$\Delta D^{B}=|D_{kuj}{}^{B}-D_{kxj}{}^{B}| \tag{6}$$

通过式(3)，可以对于下载速度的信任值 T^{jB} 进行评价，T^{jB} 的值越大，说明 j 提供的服务的可信度就越高。同时，这里考虑了个体节点经验差异的问题，通过引入相对经验因子的方法，对反馈可信度进行更新，更能体现节点的个性化特性，提高了信任评价的准确性。

同理，可以对节点 j 的 T^{jC},T^{jD},T^{jE} 进行评价，验证的方法可以采用文献[3,10,21]等算法。针对不同的声明的特点，可采用不同的信任评价算法，以提高信任评价的准确性。限于篇幅，具体算法不予讨论。

通过如上的准确性验证算法，可以得到对节点 j 发布的 j^0 类服务声明的各个因素的整体评价，评价值表示为六元组 $T^j=(j^0,T^{jA},T^{jB},T^{jC},T^{jD},T^{jE})$，节点 j 声明的各项值越高，说明节点 j 越可信。这里存在着节点 j 的某一项声明的信任评价值很高，而另一项声明如信任值却很低的问题，通过六元组 T^j 的方法记录信任的评价值，能提高信任评价的粒度。

此时访问节点在选择服务时，能看到对服务节点评价各方面因素的信任值，以下载速度为例，访问节点可以通过存放信任评价信息的档案点查看到服务的评价表，如表 3 所列。同理，也可以查看到声明 j^A,j^C,j^D,j^E 的相关列表，限于篇幅，这里就不再列出。

表 3　访问节点对 j 的下载速度的信任评价表

下载速度(jB)	提供服务的节点	对节点的下载速度的评价
$j^B\geqslant$1Mbps	$j_{m1},j_{m2},j_{m3},j_{m4},j_{m5}$……	$T^{jm1}{}^{B},T^{jm2}{}^{B},T^{jm3}{}^{B},T^{jm4}{}^{B},T^{jm5}{}^{B}$……
512kbps$<j^B<$1Mbps	$j_{n1},j_{n2},j_{n3},j_{n4},j_{n5}$……	$T^{jn1}{}^{B},T^{jn2}{}^{B},T^{jn3}{}^{B},T^{jn4}{}^{B},T^{jn5}{}^{B}$……
128kbps$<j^B\leqslant$512kbps	$j_{x1},j_{x2},j_{x3},j_{x4},j_{x5}$……	$T^{jx1}{}^{B},T^{jx2}{}^{B},T^{jx3}{}^{B},T^{jx4}{}^{B},T^{jx5}{}^{B}$……
64kbps$<j^B\leqslant$128kbps	$j_{y1},j_{y2},j_{y3},j_{y4},j_{y5}$……	$T^{jy1}{}^{B},T^{jy2}{}^{B},T^{jy3}{}^{B},T^{jy4}{}^{B},T^{jy5}{}^{B}$……
$j^B\leqslant$64kbps	$j_{z1},j_{z2},j_{z3},j_{z4},j_{z5}$……	$T^{jz1}{}^{B},T^{jz2}{}^{B},T^{jz3}{}^{B},T^{jz4}{}^{B},T^{jz5}{}^{B}$……

通过表 3 可知，访问节点可以看到其他节点对该节点下载速度的评价，有利于访问节点根据自身的网络带宽选择服务，能根据其他节点对提供服务节点的信任评价，选择高信任度的节点所提供的服务。

假设访问节点 i 是非常看重下载速度的节点，节点 i 想得到 j^0 类服务，节点 i 首先通过信任评价信息的档案点查看表 3 所列的列表，然后节点 i 根据自身的网络带宽，选择一些适合自己下载速度的节点 $j_x(x=1,2,3,4\cdots\cdots)$，节点 i 对这些节点进行综合信任评价，综合信任评价值 R_{ij} 表示为：

$$R_{ij_x}=\mu\cdot WD_{ij_x}+(1-\mu)\cdot(T^{j_xA}\,T^{j_xB}\,T^{j_xC}\,T^{j_xD}\,T^{j_xE})(w_Aw_Bw_Cw_Dw_E)^{T} \tag{7}$$

其中，WD_{ij_x} 表示为节点 i 对服务节点 j 的直接信任整体评价，通过式(1)获得；μ 表示信任评价的采用比例，这里取 $\mu=2^{-(m_0/m_{ij_x})}$；m_{ij_x} 表示节点 i 与节点 j_x 直接交互的次数；m_0 表示节点是否有经验的门限值；μ 也可以通过用户自定义的方式获得。

通过式(7)，节点 i 可以对适合自己需求的节点进行综合信任评价，采用综合信任值高的节点进行交互或合作，这样可以避免服务选择时的盲目性，能够提高服务选择的效率和交互的成功率。

4 仿真实验

假设服务节点 j 是一个稳定的节点，节点 j 提供 j^0 类服务，假设 j 所提供的服务无病毒或恶意插件，平均数据包下载速度、平均服务相应时间、平均无故障相应时间及提供服务的时间段在实验过程中保持不变。

假设节点节点 k_1 与节点 j 有丰富的直接交互经验，对 j 提供服务的各个方面的评价如表 4 所列。

表 4 访问 i 和推荐节点 k_1 分别对 j 的信任评价

k_x 对 j 的信任评价	$j^0, Dk_xj^A, Dk_xj^B, Dk_xj^C, Dk_xj^D, Dk_xj^E$
k_1 对 j 的信任评价	j^0,100%,80%,80%,65%,65%
i 对 j 的信任评价	j^0,100%,80%,80%,65%,65%

设节点 k_1 是一个非常看重服务的平均无故障相应时间及提供服务的时间段的节点，通过式(1)，节点 k_1 对 j 提供服务的整体信任评价值是 70%，并且真实向其他节点进行推荐。

设访问节点 i 对 j 提供服务的各个方面的评价与节点 k_1 具有相同直接信任评价值，i 对 j 所提供的服务进行访问时，参考了节点 k_1 对节点 j 推荐信任评价，i 根据自身的交互经验对节点 k_1 的反馈可信度进行更新。

与节点 k_1 不同的是，节点 i 非常看重数据包的下载速度和平均服务相应的时间，并通过式(1)对 j 提供服务的整体信任评价。这里的 i 是一个善意节点，提供真实的评价。

这里设计了两种模式，一种模式是采用多元组的方式存放评价信息，这里称之为考虑了信任评价的粒度；另一种模式是不采用多元组的方式存放评价信息，这里称之为未考虑信任评价的粒度。图 1 显示了两种模式下节点 i 对节点 k_1 的反馈可信度进行的更新。这里假设节点 i 对 k_1 的反馈可信度的初始值为 0.5。

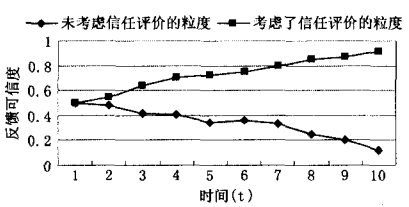

图 1 节点 i 对节点 k_1 的反馈可信度的更新

从图 1 可以看到，在未考虑信任评价粒度的模型中，节点 i 对有经验的真实的节点 k_1 的反馈可信度更新，不仅没有增加反而降低，这跟事实不符，这种错误是因为访问节点跟推荐节点对服务评价时，评价的角度不同造成的。本文所提出的多粒度信任模型，避免了这种错误，提高了信任评价的准确度。

结束语 为了解决分布式网络中，节点进行信任评价时，有的节点看重下载速度而有的节点则更看重服务质量等方面的主观差异问题，及现有信任模型对信任的评价比较笼统，信任评价比较粗糙的问题，本文通过引入经验因子的方法和采用多元组的信任信息记录方法，提出了一种基于个体经验的多粒度信任模型。实验分析表明，新模型精化了信任算法的粒度，提高了信任评价的准确度，在某种程度上消除了个体节点信任评价差异所产生的影响。

参 考 文 献

[1] Beth T, Borcherding M, Klein B. Valuation of trust in open network[C]// Gollmann D, ed. Proc. of the European Symp. on Research in Security (ESORICS). Brighton: Springer-Verlag, 1994:3-18

[2] Beth T, Borcherding M, Klein B. Valuation of trust in open networks[C] // Proceedings of the European Symposium on Research in security. Brighton: Springer-Verglag, 1999:59-63

[3] Dempster A P. Upper and Lower Probability Induced by a Multi - valued Mapping[J]. Annals Mathematical Statistics, 1967, 38(2):325-339

[4] Shafer G. A Mathematical Theory of Evidence. Princeton[M]. Princeton University, 1976

[5] Mui L, Mohtashemi M, Halberstadt M. A computational model of trust an reputation[C]// Proceedings of the 35th Hawaii International Conference on System Sciences. 2002

[6] 唐文，陈钟. 基于模糊集合理论的主观信任管理模型研究[J]. 软件学报，2003，14(08)：1401-1408

[7] Mekouar L, Iraqi Y, Boutaba R. A Reputation Management and Selection Advisor Schemes for Peer-to-Peer Systems[C]// 15th IFIP/IEEE International Workshop on Distributed Systems: Operations & Management. CA, USA, 2004

[8] Kamvar S D, Schlosser M T, Garcia-Molina H. The Eigentrust Algorithm for Reputation Management in P2P Networks[C]// Proceedings of the 12th International World Wide Web Conference. Budapest, Hungary: ACM Press, 2003:640-651

[9] Yamamoto D, Asahara, Itao T, et al. Distributed Page rank: A distributed reputation model for open P2P networks[C]// Proceedings of the 2004 International Symposium on Applications and the Internet Workshops. 2004

[10] 田慧蓉. P2P 网络信任模型与激励机制的研究[D]. 北京：北京邮电大学，2006

[11] Cornelli F, Damiani E, Vimercati D C, et al. Choosing reputable servants in a P2P network[C]// Lassoer D, ed. Proc. of the 11th Int'l World Wide Web Conf. Hawaii: ACM Press, 2002:441-449

[12] Wang Y, Vassileva J. Trust and Reputation Model in Peer-to-Peer Networks[C]// Third International Conference on Peer-to-Peer Computing. IEEE, 2003:01-03

[13] Yu B, Singh M P, Sycara K. Developing Trust in Large-Scale Peer-to-Peer Systems[C]// Proceedings of the 1s' IEEE Symposium on Multi-Agent Security and Survivability. Philadelphia, 2004

[14] Song S, Hwang K, Zhou R F. Trusted P2P Transactions with Fuzzy Reputation Aggregation[J]. IEEE Internet Computing, 2005:18-28

[15] Sabater J, Sierra C. Reputation and social network analysis in multi-agent systems[C] // First International Joint Conference on Autonomous Agents and Multi-Agent Systems. Bologna, 2002

(下转第 105 页)

的方案。也就是在每一轮加密的过程中，同时进行下一轮密钥的扩展，这样设计的好处在于减少了存储器的需求。这种方案对于加密是最优的，但是对于解密运算不可行。解密的密钥扩展过程虽然与加密相同，但选取的次序正好相反，而密钥扩展过程是不可逆的，因此我们采用在算法设定阶段就一次性完成密钥扩展的设计方案，依据迭代中 AddRoundKey 的次数，一次性生成全部轮密钥，并保存起来。这种方式虽然占用比较多的存储器资源，但针对一个同时实现加密、解密两种功能的算法，也是唯一的设计方案。

4.4 迭代轮数的划分

在设计之初，考虑用 AddRoundKey 来划分迭代次数，也就是传统意义的轮数。在每一轮中根据外部参数来连线 SubBytes，ShiftRows，MixColumns 和 AddRoundKey 4 种不同的运算模块的输入输出，但由于参数的不确定性，4 个模块之间的连线剧增，直接导致最终实现的统一框架时钟频率不可行。综合考虑，最终的方案在一个时钟周期内完成一种运算模块。

对于标准 AES 加密，这里需要 42 个时钟周期完成一个分组的加、解密。

4.5 实验结果

以 Altera 公司的 Cyclone II 系列芯片 EP2C70F672C8 作为算法载体，通过 Verilog 编程实现了类 AES 算法统一框架，并利用 ModelSim 对其进行了仿真。

实验结果为 SubBytes 模块占用 1362 个逻辑单元，ShiftRows 模块占用 130 个逻辑单元，MixColumns 模块占用 391 个逻辑单元，AddRoundKey 占用 128 个逻辑单元。统一框架共实例化 20 个 SubBytes 模块(加密、解密运算部分并行实例化 16 个 SubBytes 模块，密钥扩展部分并行实例化 4 个 SubBytes 模块)、1 个 ShiftRows 模块、4 个 MixColumns 模块、1 个 AddRoundKey 模块，使用 59 个 I/O 引脚，共计 41007 个逻辑单元，时钟频率为 66MHz。以标准 AES 算法加密的迭代次数为例，加/解密使用 42 个时钟周期，吞吐率为 201Mbits/s。

5 安全性分析

我们给出的基于 AES 算法的统一框架模型，是采用 SP 结构，直接依据 Shannon 提出的混淆和扩散原则设计的。本文着重从工程实现的角度考虑，从最一般的角度实现 AES 算法组件评估，进而估算硬件实现效率。

算法的安全性由选取合适的算法组件及迭代次数保证，根据选用的算法组件密码学特性，通过密钥给出足够的迭代次数。算法安全性由算法组件的安全性分析驱动，通过外部密钥的设定，在不更改电路结构的情况下，灵活适应最新的密码学研究成果。

结束语 本文依托 Kerckhoffs 假设，创造性地将密钥扩展为算法密钥和数据密钥。通过将 AES 算法模块化，以牺牲一定资源为代价，在保证算法实现效率的前提下，兼顾了算法的灵活性，一定程度上提高了算法使用的安全性。

文中给出了类 AES 算法最一般的元素部件硬件资源评估。在实际应用中，如何快速、高效地找出密码学性质好的运算，进而利用该类运算特点优化硬件实现，提高算法效率，还需要进一步做工作。

参 考 文 献

[1] Schneier B. Applied Cryptography[M]. Second Edition: protocols, algorithms, and source code in C. New York: John Wiley & Sons Inc, 1996

[2] 高娜娜，王沁，李占才. 基于 AES 和 DES 算法的可重构 S 盒硬件实现[J]. 小型微型计算机系统，2006(3)：446-449

[3] Zhang X, Parhi K K. An efficient 21.56 Gbps AES implementation on FPGA[C]// Thirty-Eighth Asilomar Conference on Signals, Systems and Computers. Nov. 2004, 1: 465-470

[4] Tim G, Mohammed B. AES on FPGA from the fastest to the smallest [C]// Proceedings of CHES 2005. Springer, 2005: 427-441

[5] Gael R, Francois-Xavier S, Jean-Jacques Q, et al. Compact and efficient encryption/decryption module for FPGA implementation of the AES Rijndael very well suited for small embedded applications [C]// Proceedings of the International Conference on Information Technology: Coding and Computing. 2004, 2: 583-587

[6] 冯登国，吴文玲. 分组密码的设计与分析[M]. 北京：清华大学出版社，2000

[7] Daemen J, Rijmen V. The Design of Rijndael: AES- the Advanced Encryption Standard[S]. Springer-Verlag, 2002

(上接第 94 页)

[16] Liang Z Q, Shi W S. PET: A Personalized trust model with reputation and risk evaluation for P2P resource sharing[C]// the 38th Hawaii International Conference on System Science. 2005

[17] Lee S, Sherwood R, Bhattacharjee B. Cooperative peer groups in NICE[C]// IEEE Infocom. San Francisco, USA, 2003

[18] Stoica I, Morris R, Karger D, et al. Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications[C]// SIGCOMM 2001. San Diego, California, USA, 2001

[19] Aberer K, Cudre-Mauroux P, Datta A, et al. P-Grid: A Self-organizing Structured P2P System[J]. SIGMOD Record, 2003, 32(3): 29-33

[20] Ratnasamy S, Handley M, Karp R, et al. Application-level multicast using content-addressable networks[C]// Proceedings of Third International Workshop on Networked Group Communication. 2001: 14-21

[21] 张蓉，张霞，文学志，等. Peer-to-Peer 环境下多粒度 Trust 模型构造[J]. 软件学报，2006, 17(1): 96-107

[22] Ratnasamy S, Shenker S, Stoica I. Routing algorithms for DHTs: Some open questions[C]// Druschel P, ed. Proc. of the 1st Int'l Workshop on P2P Systems. Berlin: Springer-Verlag, 2002: 45-52