一种基于社交关系模型的系统安全分析方法

向 坚[1], 刘 [1], YU Eric[2]
(1. 清华大学软件学院, 北京 100084; 2. 多伦多大学信息学院)

摘 要: 安全问题已经成为现今信息技术在企业应用中要解决的重要问题, 本文提出一种设计安全系统的方法框架, 这个方法基于面向主体的需求建模框架 i^*, 通过社交概念模型来分析与安全相关的系统的业务和组织环境. 这个方法框架把安全分析和一般的软件工程分析方法结合起来, 把安全性和系统相关的其他功能性需求和非功能性需求的共同分析, 对需求进行权衡, 从系统设计的初始阶段起就把安全措施整合进去. 本文通过网上购物的实例用 i^* 图示分析了设计过程中的相关步骤.

关键词: 安全设计; 社交关系模型; i^* 框架; 软件工程
中图分类号: TP311 文献标识码: A 文章编号: 0372-2112 (2006) 12A-2350-05

Security Design Based on Social Modeling

XIANG Jian[1], LIU Lin[1], YU Eric[2]
(1. *School of Software, Tsinghua University, Beijing* 100084, *China*; 2. *Faulty of Information Studies, Toronto University, Toronto, Canada*)

Abstract: We propose a methodological framework for designing security systems. This framework is founded on agent-oriented i* requirement modeling framework. We propose to use social modeling concepts to analyze the business and organizational context of systems with regard to security. This methodological framework encompasses analysis on the functional and non-functional requirements in relevance to security, making trade-off when design, thus integrating security into the system design process from the outset. This paper uses the Online Ordering example for illustration.

Key words: security design; social modeling; i* framework; software engineering

1 引言

在当前开放网络环境下, 软件系统的安全设计变得尤为重要和复杂, 成为一个广受关注的问题. 安全设计需要全面理解系统环境中的各种依赖关系, 因此我们需要通过系统化的步骤来确定相关的主体、认识各个相关决策者的偏好、计划, 权力及依赖制约关系. 进行安全设计前必须确认需要保护的对象、确认在被攻击时可能会失效的那些系统的薄弱点. 除此之外, 预测可能的攻击者和他们的动机、能力以及行为也是十分重要的. 只有从攻防双方进行全面的分析, 才能对安全需求进行有效的权衡, 提出最合理的安全解决方案.

安全考虑的是攻击者有目的的行为, 而这些行为是不被保护者所认可的行为. 安全的目标是保护有价值的对象, 包括物质、信息这类物理资产以及身份、名誉等无形资产. 提供保护的机制被称为防御措施, 我们从对攻击的检测、响应、减轻、灾难恢复和反击这些方面来评价各个具体防御措施的有效性. 过度的保护和过分的信任都是不可取的, 客户真正需要的是一个“适度”的保护程度[3].

安全设计从实质上讲是一个权衡各种需求因素的过程. 在进行安全设计的决策时, 设计者不仅仅需要考虑技术相关的安全问题, 还需要考虑其他非技术因素: 如管理企业的规章制度和法律条款、参与人员在费用、性能、易用性方面的要求等等. 一般来说安全设计需要回答以下问题: “谁是组织中主要的参与者? 他们的业务目标是什么? 对于这些业务目标的攻击可能有哪些? 这些攻击对于业务目标的影响是什么? 可能的后果和其严重性怎样? 有哪些可以选择的方式能够控制和反击这些攻击? 怎么根据它们对商业目标的影响来对这些方式进行排序? 增加这些控制和防御手段会对原有系统产生什么影响? 能不能满足所需要的安全水平?”.

为了把上述问题和安全设计结合在一起, 有利于提出具体的解决方案, 本文提出一种设计安全系统的方法框架, 这个框架基于社交关系的相关概念, 包括参与者、角色、目标、任务、资源和主体之间的依赖关系等. 基于该框架可以从用户、管理员和设计者角度分析系统所面临的不同潜在攻击和薄弱点, 寻找和评估可用的防御手段, 以及整合不同技术以获得期望的安全水平.

收稿日期: 2006-09-22; 修回日期: 2006-11-20
基金项目: 国家自然科学基金(No. 60503030); 清华信息科学与技术国家实验室基础研究基金; 国家 973 重点基础研究计划(No. 2002CB312004)

本文采用基于角色的分析方法分析潜在的攻击. 基于角色建模一般用于分析不同抽象层的用户职责[5]. 在这里, 基于角色建模方法用于分析具体某个角色潜在的攻击问题, 例如信任关系和攻防关系. 本文的安全设计方法把软件工程中的需求分析思想和安全设计结合起来, 将安全当作众多非功能需求之一来看待, 因此安全必须从软件工程的最早期的阶段就开始明确化和进行相应的分析和处理[4]. 社交关系建模方法是源于软件工程中面向主体的观点, 而安全问题源于人类的目的性和利害关系, 所以可以用社交关系概念建模[2]. 本文将这些社交关系概念扩展到覆盖软件系统和组件之间的关系. 面向主体的模型有助于我们使用更多基于互联网环境的描述和分析技术. 面向主体的建模我们使用了典型的 i^* 需求建模框架[9], 它是十分适合早期需求建模的面向主体建模框架.

2　基于社交关系模型的系统安全设计

设计系统安全需要一个系统化和集成的方法. 为了全面理解系统安全需求和寻找一个可行的解决方法, 本文考虑采取以下步骤:

(1) 确认参与者和他们的目标

为分析信息系统的安全问题, 我们需要理解系统和系统环境. 通过清楚地表示出系统参与者、参与者通过系统完成的工作、以及参与者之间的关系, 我们能够为安全分析构造基本的系统结构.

(2) 确认可能的攻击者

设计安全必须从认识系统可能的攻击者开始. 如果我们不知道攻击者和一般的攻击方式, 很难为系统增加有效的控制;

(3) 评估攻击的结果

防御所有的攻击是不太可能的, 或者说由于过于昂贵而不现实. 通过评估受到攻击的后果, 我们为潜在的攻击进行优先级排序, 然后利用优先级顺序纳入安全设计.

(4) 确认可选的防御手段和选择解决方案

通过选择可用的安全措施, 增加精确的保护过程、策略和技术, 我们能够构建出一个比较理想的设计方案, 需要注意的是这个设计方案在解决安全问题的同时还要考虑其他功能性和非功能性需求.

上述几项是本文提出的分析框架的基本步骤, 下面逐一进行分析, 举例.

3　确认参与者和其业务目标

安全设计的主要目标是保护系统不受潜在的破坏, 但同时我们必须支持系统中相关参与人员的业务职责以及潜在意图, 就如同我们不可能为了防止空难而不坐飞机. 因此, 我们首先要知道, 谁参与了这个系统? 参与者主要的业务目标是什么? 参与者之间的关系是什么? 回答这些问题需要建立各个功能单位之间的概念模型. 我们寻找组织中参与者的基本动机, 和参与者之间的决定性依赖关系, 而不是像通常的建模方法中的分析活动和流程. 从这种策略性的观点出发, 我们就理解为什么系统需要某种需求, 然后决定系统应该怎么满足这些需求.

主要的参与人被表示为参与者. i^* 中的参与者一般用于表示具有目的性和社交依赖关系的单元. 把系统单元看作独立的参与者很适合对当前的开放分布式组织建模. 在 i^* 中, 参与者进一步细分为角色, 代理和职位. 角色是一个具有期望和职责的抽象参与者; 代理是一个具体的参与者, 人或者机器, 具有具体的能力和功能. 一个代理能够担任多个角色. 而职位是一系列的角色封装在一起, 可以分配给某个具体代理.

用 i^* 建模的重点在于分析主体的目的性和理性, 以及社交依赖关系. 不同于其它建模框架之处在于, 其他建模框架重点在于信息交换或是控制流程. 通过在策略依赖关系模型中对于依赖关系的分析, 我们能够推理出社交网络中潜在的机会和脆弱点. 依赖关系的类型用来区分关系参与者在决策和行为方面的自由度, 目标依赖关系中, 被依赖的参与者有最大的自由度来决定怎么满足这个依赖关系; 任务依赖关系中的被依赖方则必须遵循一些由依赖方约定的行为; 而资源依赖是被依赖方提供给依赖方可用的资源; 软目标依赖类似于目标依赖关系, 但是由依赖方来决定依赖关系被充分满足与否.

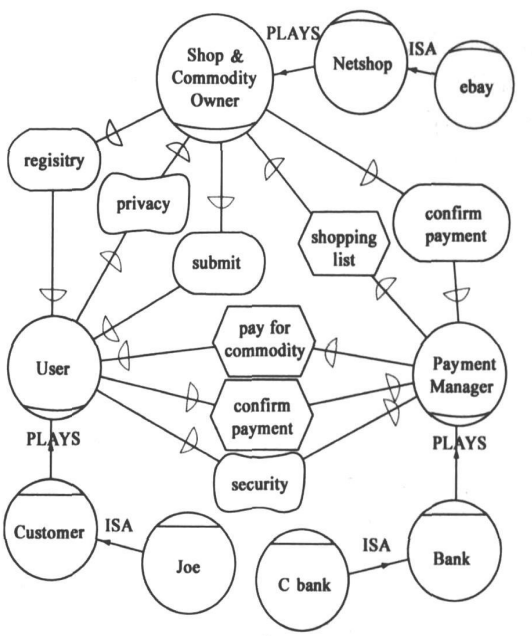


图 1　网上购物社交依赖模型

图 1 中的 i^* 社交依赖模型说明怎样确认系统的参与人员和业务目标. 这是一个网上购物系统的实例. 这个模型显示, 网上购物商店(Netshop)目标依赖用户(User)注册使用, 同时用户依赖网上商店保护自己个人的隐私, 隐私属于软目标; 银行(Payment Manager)任务依赖用户付钱(pay for), 而用户依赖银行保护交易的安全性, 安全作为软目标考虑; 银行任务依赖网上商店提供用户提交的购物单, 而网上商店目标依赖银行确认用户已经付钱. 图中针对各个角色都有对应的代理来担任, 如用户(User)这个角色是由客户(Customer)来担任的, 而一个具体的人(Joe)是一个(ISA)用户. 通过这个模型的建立, 我们能清楚地分析系统的业务相关内容, 有利于后续的攻

防分析.

4 攻击者和攻击方式确认

4.1 攻击者分析

攻击者是那些有意进行不被系统认可的行为的参与者. 各种可能的动机和可以利用的资源使得人成为最危险的攻击者. 我们进行攻击者分析的目标是找出那些潜在的滥用系统的人员和他们恶意的攻击目的. 本文提出的安全分析方法中, 我们作"有罪假设", 即首先假定所有的参与者都是被潜在攻击者, 除非有充分证据证明他们不会攻击系统. 在确认攻击者时, 我们需要沿着某些维度上设定一组信任边界(外边界, 内边界), 在外边界范围以外的攻击会被认为是不可能发生的, 内边界范围内的攻击被忽略, 即在内部边界内部的参与者是被信任的. 设定必要的边界可以划定问题的范围. 对于组织外部所有的参与者我们都会进行逐一分析, 这些参与者能够以何种方式攻击系统以及他们如何从这种攻击中获益. 我们分析攻击者, 是把每一个拥有部分系统资源的参与者作为分析对象, 这个参与者如果存在恶意的情况下, 会对系统产生什么样的攻击, 可能导致何种后果.

攻击者可以根据不同的标准分类, 包括动机、目标、特殊技术、资源和风险. 在我们的攻击者分析中, 就每一个参与者进行分析, 而每个参与者都占有一定的资源和具有一定的行为能力, 其目标和动机我们都应加以考虑.

4.2 攻击方式分析

在对攻击者身份角色分析之后, 我们进行攻击方式分析. 这里是从基于角色的角度来分析, 每个攻击者占据一个或多个角色, 并且继承这些角色对应的合法用户的所有目标、能力和社会关系(内部目标和外部的依赖关系)[6]. 我们从两个视角来分析安全, 一个是正向分析, 即对于所有的参与者开始分析, 看他们利用本身角色所拥有的资源和能力能够完成什么样的恶意攻击; 另一个是反向分析, 即从可能的恶意目的和被攻击的对象开始分析, 寻找潜在的攻击者和攻击路径. 从两个角度同时分析, 能对潜在的威胁有一个全面的把握.

对于潜在攻击的分析是十分关键的, 它将影响到系统的安全性高低, 期望建立完美的安全系统是不现实的, 所以我们对于可能的攻击方式应该有一个合理的优先级排序, 最可能出现、最紧急、最需要解决的安全问题是什么, 有的攻击频繁出现, 但影响很小; 有的攻击很少出现, 但可能导致系统完全瘫痪. 只有充分分析潜在攻击, 才能给我们的系统安全措施和总体设计提供最大的帮助.

把每一个参与者都作为攻击者看待, 我们从下面几个方面考虑:

(1) 这个参与者拥有什么样的资源、能力? 利用这些资源和能力, 他能做什么?

(2) 这个参与者和其它哪些参与者有什么样的依赖关系?

(3) 如果这个参与者是一个攻击者, 他能对哪些对象发起何种类型攻击?

(4) 这个攻击会导致什么样的后果? 通过和其它参与者的依赖关系, 这个攻击后果会对系统带来什么样的连锁反应?

在这个模型中, 攻击的效果能够通过模型建模和分析出来. 将所有系统的参与者都分析一次, 能够对系统潜在攻击者的情况有一定认识, 有利于安全设计时攻击手段的排序和防御手段的选择, 在后续工作中, 如果把这个模型的各个参与者内部细化, 建出目标分析模型, 我们能更加清楚地分析出攻击者产生攻击的途径、后果以及连锁影响.

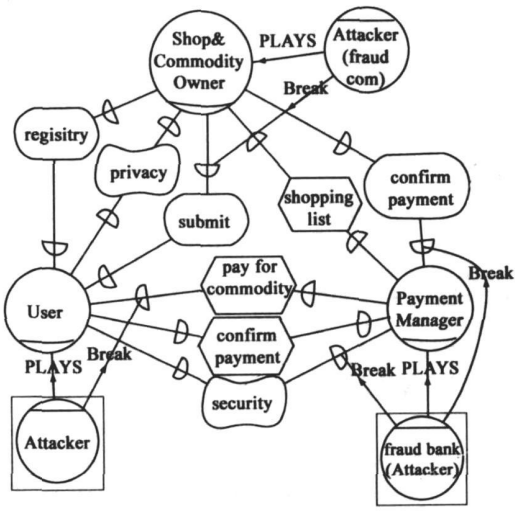


图 2 网上购物攻击者分析模型

在图 2 中, 我们将每个参与业务的参与者都作为一个潜在的攻击者进行分析, 在这里主要体现为网上商店攻击者 Attacker(fraud netshop), 银行攻击者 Attacker(fraud bank), 网络黑客攻击者 Attacker(hacker user), 这些攻击者可能造成的负面影响关系也在图中表示出来, 如当把用户(user)作为攻击者(attacker)考虑时, 这个攻击者(attacker)可能提交订单但是不付钱, 或者使用假的银行卡、或者伪造身份, 这些都会对于任务依赖关系(pay for commodity)产生破坏影响; 类似的, 作为攻击者(attacker)分析的银行, 可能破坏和网上商店之间的信息准确性(accuracy), 也可能破坏和用户之间的信任关系(trustworthiness). 通过这个模型的分析建立, 我们对于潜在攻击有较为全面的把握, 需要注意的是, 攻击分析不可能十分完整, 所以我们的模型总是可以也需要不断扩展.

5 确认候选防御手段和选择合理的解决方案

通过前面的攻击分析, 我们找出了潜在的攻击, 而对于某一类的攻击, 我们需要选择不同的保护措施, 这些措施在功能性需求和非功能性需求例如效率、费用、易用性等的影响肯定会有不同, 我们需要做的, 是均衡地判断一种合理的措施, 能够全面满足我们的要求, 并不是片面地追求绝对安全.

成功的攻击有三个必不可少的因素, 分别是攻击者的动机, 系统的脆弱性和攻击者进行攻击的能力. 因此, 为了对抗一个潜在的攻击, 我们应该寻找适当措施来防止或者抵消上述三个因素之一或者全部. 在防御手段分析过程中, 系统的设计者和系统参与人讨论决定怎么保护系统不受潜在攻击者的攻击. 这个分析过程覆盖一般性的攻击, 然后有选择性地把原型的解决方案应用到不同参与者的不同具体需求, 得到一个

整体的解决方案. 一般类型的攻击和原型的解决方案通过分类法和安全方面的知识库得到[2].

保护者一般是通过面向目标的分析来寻找针对某个攻击的防御手段, 面向目标即怎样解决某个攻击. 对于每一个确定的攻击, 一般是在评估了它对于系统和业务目标影响以后, 再考虑阻止和保护措施. 评估攻击的风险后, 我们应做出必要的抉择, 一是宏观的选择, 是接受风险而不做任何事, 还是在设计中增加防范控制措施; 同时我们也应该针对问题作出具体的选择, 是消除风险动因, 还是消除风险后果, 还是最小化脆弱性, 还是控制不利影响.

对于保护者, 攻击在建模中被表示为信念, 因为攻击来自外部实体, 这些资源是不受保护者控制的. 信念源自一方对于另一方的认识, 保护者的信念将决定它的防御手段. 攻击者和保护者各自有各自的信念, 保护者在自己的信念下, 寻找针对性技术和不同厂商的产品或者相关领域专家的方案, 然后这些技术, 产品和方案会被一起评估, 评估的因素包括它们能解决的攻击、开销、性能、用户友好性等等, 多数是非功能性需求. 在可能的解决方法都被评估了以后, 分解、设置和组合这些解决方法而形成最终的设计解决方案. 把安全和其他需求因素一起考虑, 均衡得出解决方案[1].

对于一个具体的安全技术、方案或者防御手段, 我们可以采用以下五个步骤分析、评估:

(1) 要保护的是什么对象?

(2) 这些对象面临了哪些风险?

(3) 这个保护措施怎么减轻攻击风险?

(4) 增加这个保护措施会给系统带来其他哪些额外风险?

(5) 保护措施会花费我们什么代价?

每个安全措施都有必要的代价, 很多安全措施只是花费金钱, 也有一些安全措施会影响到其他的非功能性需求, 例如隐私性、易用性等等, 我们需要考虑增加这个措施带来的不利影响, 也就是这个安全措施值不值得相应的代价[1].

通过上述方法可以评估防御手段的有效性, 我们能够确定是否攻击的影响已经减低到能够接受的水平. 由于增加保护防御手段可能会给系统带来新的脆弱点, 防御手段的分析过程会反复迭代进行, 直到找到满意的解决方法. 这样的过程可能会使解决方案的选择空间变得非常大, 但是经验告诉我们, 这种基于目标的推理方法, 能够解决常见规模的模型(含 300 左右个节点).

6　实施案例分析

在这里我们将具体分析一下这个案例在本文所提出的基于社交关系的安全设计方法下的实施过程.

Step 1: 确认参与者和他们的目标. 在图 1 中, 我们确定了网上购物过程中相关的参与者以及他们的目标, 如图 1 所示, 这里的参与者有很多, 具体的的有客户、银行和网上商店等等. 这些参与者的目标体现在他们相互之间的依赖关系上, 网上商店目标是更多的注册用户, 更多的人购买商品; 客户的目标是购买商品; 而银行的目标是准确安全地完成交易. 在网上购物这个场景中, 有很多的依赖关系, 在图 1 中只是简略地展示了一部分. 第一步确认参与者和相关业务目标实际是对业务关系的分析, 是社交关系的集中体现, 通过这一步, 安全分析人员掌握了被分析系统的基本工作信息.

Step 2: 确认可能的攻击者. 在图 2 中, 我们进行了攻击者分析. 攻击者分析是针对所有系统的参与者进行遍历分析, 把所有的攻击者当作潜在的攻击者, 这些攻击者拥有所有它所对应的参与者的能力和资源. 作为客户来说, 他们拥有注册、提交购物清单、付款购物的与其它客户同等的权利和资源. 但作为攻击者的他们可能会破坏体系中的购买依赖关系, 从而造成交易无法完成, 从客户这个角色分析出发, 可能造成虚假客户, 拒绝服务攻击等很多潜在攻击, 而银行如果作为攻击者, 会给网上商店的信任度和收益直接造成损失; 当进行具体的攻击方式分析时, 可以结合已有的安全分析数据库来考虑.

Step 3: 评估攻击的结果. 在罗列了所有的可能攻击方式以后, 为了使得防御系统更加有针对性, 我们应该对攻击风险进行排序. 这个排序的过程主要依据安全专家们的经验判断, 以及系统本身的安全需求重点. 类似风险管理, 我们需要对攻击方式进行风险后果评估. 有的攻击方式带来的后果是灾难性的, 会直接导致系统崩溃, 这是优先级比较高的攻击方式, 还有一些攻击方式是频发的, 也需要优先处理. 如针对网上商店的拒绝服务攻击(DoS), 是一个危险同时频繁的攻击方式, 需要排在较高的优先级. 这一步的最终结果是攻击方式风险由高到低的列表.

Step 4: 确认可选的防御措施和选择解决方案. 前面已经提到, 针对某一种攻击方式, 可能会有不同的防御手段, 我们将选择针对被分析系统最合理的防御措施, 防御措施必须从系统的多个角度出发考虑, 取得功能性和非功能性需求上的均衡, 如针对客户伪造身份的攻击方式, 我们可能的防御措施有身份认证、访问控制等多种选择, 但是身份认证会要求建立用户身份和密码的数据库, 访问控制需要建立权限和用户信息的数据库, 而像指纹识别之类的认证方式, 虽然相对更安全, 但是会带来很高的成本需要, 我们将选择最均衡的措施来解决攻击问题, 并非最安全, 而是最合适的安全措施. 因此, 作为不同的系统要求, 这种分析方法会给出不同答案.

从案例实施的角度来说, 要成熟地使用这个安全分析方法, 还需要解决过程中的许多问题, 包括攻击方式的风险排序, 防御措施的权衡评估等, 都是需要在实践中逐步完善的工作.

7　相关工作和讨论

传统的风险分析和管理工具, 通常提供一个可扩展的数据库, 这个数据库中包含了可能的攻击方式、资产和控制方式, 安全分析者可以选择它的一个子集. 本文方法的目标是建立一个通用的安全相关设计知识的分类, 这个分类中包含了传统的数据库内容, 但是不仅限于此. 主要的区别在于我们分析安全设计的同时, 考虑其他的功能性和非功能性需求, 因此这些需求之间的交互也在考虑和解决范围之内[1].

许多相关的工作通过攻击树[8]或者其他的模型来获得安全需求和开发安全系统, 这些方法在具体的设计阶段是十分有效的安全分析技术, 我们可以把框架和这些技术在不同层

次整合起来, 例如, 我们使用 i^* 模型进行第一阶段的粗粒度的分析, 得到我们在安全设计中需要完成的目标, 在 i^* 中这些目标是没有操作化的, 在面向过程的操作细节的获取过程中, 这些分析技术就能派上用场.

在软件需求工程领域, 文献[4]最早提出把安全需求和其他的功能性和非功能性需求一起考虑, 从早期开始直到具体的设计阶段[4]. 本文的这个方法是符合这个观点的, 不同的是我们是把目标分析和面向主体的分析一起进行.

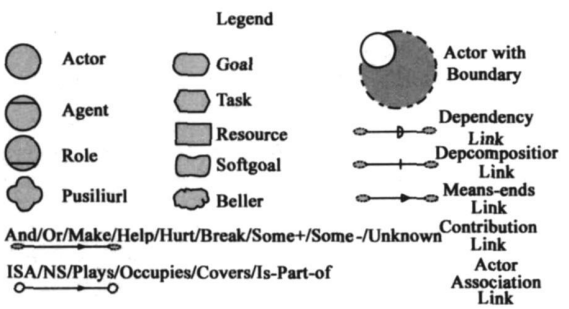
从这个方法框架出发, 我们可以用它分析现实生活中的案例, 利用案例分析确认这个框架的量化可能; 我们还可以形式化这个分析技术, 然后开发工具用于支持这种安全分析.

8 结束语

本文提出一个基于面向主体的需求分析框架的安全分析和设计的方法框架. 主要目标是定义一系列的安全相关的分析机制, 并且把它们和软件工程过程整合起来, 将安全需求和其他的功能性和非功能性需求一起考虑. i^* 框架中涉及的概念使我们能够在社交关系上下文中分析安全问题. 而面向主体和面向目标的分析方法的结合, 使得我们能够完成不同的对策分析. 这个基本方法框架能够用于自顶向下安全需求分析过程, 或时自底向上理解的过程. 基于目标的评价技术有利于安全和其他质量需求的平衡分析.

附录

以下是 i^* 建模框架的主要的元素表示:



* 文中所有模型的运用组织建模工具 ome3 创建.

参考文献:

[1] Schneier, B. Beyond Fear, Thinking Sensibly About Security in an Uncertain World[M]. Copernicus Books, 2003.

[2] Liu L, Yu E, Mylopoulos J. Security and privacy requirements analysis within a social setting[A]. The 11th IEEE Int Requirements Engineering Conference(RE'03)[C]. Monterey Bay, California USA, 2003. 8-12.

[3] Sandu R. Good-enough security: towards a pragmatic business-driven discipline[J]. IEEE Internet Computing. Security Track, 2003. 7(1): 66-68.

[4] Chung L, Nixon B A, Yu E, Mylopoulos J. Non-Functional Requirements in Software Engineering[M]. Kluwer Academic Publishers, 2000.

[5] R Falcone, M Singh, Y H Tan, et al. Trust in Cyber-Societies-Integrating the Human and Artificial Perspectives[M]. Berlin: Springer, 2001. 175-194.

[6] Yu E. Agent-oriented modelling: software versus the world[A]. Agent-Oriented Software Engineering AOSE-2001 Workshop Proceedings[C]. LNCS 2222. Springer Verlag, 2001. 206-225.

[7] Liu L, Yu E, Mylopoulos J. Security design based on social modeling[A]. Proceedings of Thirteenth Annual International Computer Software & Application Conference(COMPSAC)[C]. Chicago, 2006. 71-76.

[8] Schneier B. Attack trees: modeling security threats[J]. Dr Dobb's Journal, 1999, 12(24): 21-29.

[9] Yu E. Towards modeling and reasoning support for early-phase requirement engineering[A]. Proceedings of the 3rd IEEE International Symposium on Requirements Engineering(RE97)[C]. Washington, 1997. 226-235.

作者简介:

向 坚 男, 1983 年生于湖南沅陵县, 清华大学软件学院在读硕士研究生. 主要研究方向: Web 服务、语义网、网络安全.

E-mail: xiangj05@mails.tsinghua.edu.cn

刘 女, 清华大学软件学院副教授, 博士. 主要研究领域为需求工程、信息系统工程与基于知识的软件工程.

E-mail: linliu@tsinghua.edu.cn

YU Eric 男, 加拿大多伦多大学信息学院副教授, 博士. 主要研究领域为需求工程、信息系统工程、组织建模与分析