

基于 AMQP 消息中间件的主动 状态证书撤销机制实现*

付建丹, 熊选东, 张亮忠, 王松锋, 王 珏

(信息工程大学 电子技术学院, 郑州 450004)

摘 要: 主动状态证书撤销机制是一种新的证书撤销验证机制, 如何实现该机制还未见具体的案例。在分析高级消息队列协议(AMQP)的基础上, 结合主动状态证书的订阅/发布要求, 设计了一种基于 AMQP 消息中间件的主动状态证书发布与订阅模型。通过实验分析表明, 该消息中间件系统应用灵活, 系统性能高效、稳定。

关键词: 高级消息队列协议; 消息中间件; 主动状态证书; 证书撤销; 发布/订阅

中图分类号: TP309 **文献标志码:** A **文章编号:** 1001-3695(2012)05-1918-04

doi:10.3969/j.issn.1001-3695.2012.05.085

Implementation of active status certificate revocation scheme based on AMQP message-oriented middleware

FU Jian-dan, XIONG Xuan-dong, ZHANG Liang-zhong, WANG Song-feng, WANG Jue

(Institute of Electronics Technology, Information Engineering University, Zhengzhou 450004, China)

Abstract: Active status certificate revocation is one of the certificate revocation verification schemes, while there is not any its implementation case. Analyzing the advanced message queue protocol(AMQP) and the requirement of publish/subscribe active status certificate, this paper implemented a new active status certificate publish/subscribe model based on AMQP message-oriented middleware. By testing its performance, the application of the message-oriented middleware system is flexible and the system's performance is high efficient and stable.

Key words: AMQP; message-oriented middleware; active status certificate; certificate revocation; publish/subscribe

在 PKI 运行中, 某些证书由于私钥泄露或所有者身份变更等原因, 需要在该证书有效期内将其撤销, 并将撤销信息安全及时地发布, 以供证书用户在验证签名或进行安全通信时验证相关证书的有效性^[1]。

证书撤销列表(certificate revocation list, CRL)和在线证书状态协议(online certificate status protocol, OCSP)是目前 PKI 实施中常用的证书撤销验证机制^[2,3]。CRL 通常发布到 LDAP 服务器上, 由证书用户定期下载。OCSP 是一种在线证书状态请求/响应协议, 证书用户向 OCSP 响应器发送证书状态查询请求, OCSP 响应器实时返回所查询证书的状态信息。

文献[4]提出了一种新的证书撤销验证机制, 该机制兼具 CRL 和 OCSP 的优点, 应用灵活, 可以大幅度地降低证书撤销验证过程中的网络通信负载, 有效地减少密码运算次数。

文献[4]中对主动状态证书撤销机制实现提出了主动求证、在线验证、主动下载验证和定制服务验证四种形式, 并对前三种形式作了描述, 但对定制服务验证形式未给出具体的方法, 且都未给出实现模型。其他还未见针对主动状态证书的相关研究。

AMQP^[5]是由 JP Morgan、iMatix、Cisco、IONA、Novell、Red-hat 等机构联合开发的一个面向消息的中间件协议, 是应用层

协议的一个开放标准。目标是提供实时、低成本、可靠的中间件通信。该协议定义了网络通信协议以及建立在之上的服务。AMQP 在金融等行业广泛用于消息的发布与订阅。基于 AMQP 的消息中间件有 OpenAMQP、Apache Qpid、RabbitMQ 等产品。

文献[6]中介绍了 AMQP, 并将其开源实现 Apache Qpid 进行了性能测试, 同时与现实中的股票交易传递系统进行了性能比较, 得出了 AMQP 消息中间件在消息传递中的高效性与灵活性。

本文研究如何基于 AMQP 消息中间件实现主动状态证书撤销机制。

1 AMQP 消息中间件

AMQP 是消息中间件的开放标准, 与语言、平台无关, 是消息中间件标准化的一个具体方案^[7]。基于 AMQP 开发的消息中间件可以进行互操作, 便于今后与其他系统的集成和自身系统的升级。AMQP 是基于客户端/代理模式的, 为了实现消息中间件的互操作能力, 它定义了一套网络协议, 以及包含对代理服务描述的模式结构。2006 年 6 月 AMQP 协议 0.8 版本发布, 2009 年下半年 0.10 版本发布, 1.0 版本还在制定中。

收稿日期: 2011-09-28; 修回日期: 2011-11-08 基金项目: 国家“863”高科技研究发展计划资助项目(2008AA01Z404)

作者简介: 付建丹(1986-), 男, 湖北丹江口人, 硕士研究生, 主要研究方向为网络技术、信息安全(abc128710@sina.com); 熊选东(1965-), 男, 湖北枣阳人, 研究员, 主要研究方向为网络技术、信息安全; 张亮忠(1985-), 男, 山西临汾人, 硕士研究生, 主要研究方向为网络技术、信息安全; 王松锋(1986-), 男, 河南登封人, 硕士研究生, 主要研究方向为系统工程、Petri 网; 王珏(1987-), 男, 江苏徐州人, 硕士研究生, 主要研究方向为数据库应用。

1.1 AMQP 模型结构

一个AMQP消息系统包括发布者(publisher)、订阅者(consumer)、代理/服务器(broker/server)三个主要部分,如图1所示。在一个主机上可以部署多个虚拟主机,每个虚拟主机由交换(exchange)、消息队列(message queue)和绑定(binding)组成。交换接收来自发布者的消息,并将这些消息路由至消息队列中,路由基于消息的主题或内容;消息队列储存消息,直至这些消息能够被相应的客户端程序接收;绑定定义消息队列和交换之间的联系,并且提供消息路由规则。一个交换可以接收多个发布者发布的消息,一个消息队列可以相应接受多个订阅者的消息订阅。

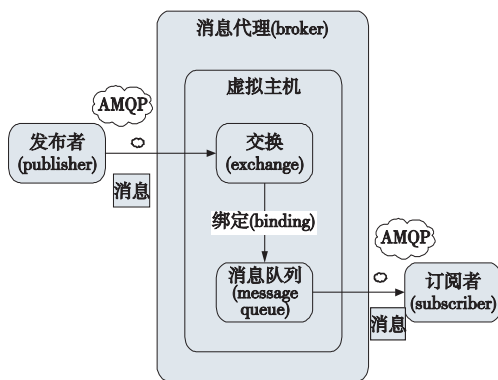


图1 AMQP模型结构

1.1.1 消息队列

消息队列将消息存储于内存或硬盘中,并且将它们顺次地传递给一个或多个消费程序。消息队列是消息的存储和传递实体,每个消息队列完全独立。

消息队列具有如私有或共享、持久或短暂、永久或暂时之类的属性。通过选择需要的属性,消息队列可实现下列中间件消息队列:

- 标准的储存和转发队列。储存和转发消息给基于时间片轮转的多个订阅者,是典型持久和共享的服务多个订阅者的队列。
- 短暂回复队列。储存和转发消息给一个单独的订阅者,是典型短暂和私有的服务单个订阅者的队列。
- 发布订阅队列。储存来自各个已订阅的数据源处的消息,并将它们转发给一个单独的订阅者,是典型的短暂和私有的服务单个订阅者的队列。

1.1.2 交换

交换接收来自消息发布者的消息,并将消息依据事先商定好的规则路由至消息队列,这些规则就称为绑定。交换作为匹配和路由引擎,通过检查消息和使用绑定表格,决定如何将消息转发至各队列中,它本身不储存消息。

AMQP定义了许多标准的交换类型,涵盖了完成普通消息转发所需要的基本路由类型,其中规定必须实现的交换类型包括direct、fanout、topic和headers类型,并且这些类型在服务器启动时均会默认生成。各种类型的工作模式如下:

- Direct类型工作模式。Message queue与exchange通过key K进行绑定,发布者以路由值key R将消息发布至exchange,消息被发至所有使key K = key R的队列中。
- Fanout类型工作模式。Message queue不作类型声明与exchange绑定,发布者将消息发送至该类型的exchange时,消息将被无条件地发至所有与该exchange绑定的队列。

c)Topic类型工作模式。Message queue与exchange通过key K进行绑定,发布者以路由值key R将消息发布至exchange,key K是以“.”分隔的字符串,如“eur.stock.db”。当key R是“*.stock.#”时,表明key K和key R是相匹配的,那么消息就会被发至该队列中。

d)Headers类型工作模式。Message queue与exchange之间的绑定通过一系列的“属性=值”的集合来表示,这些属性之间采用简单断言连接。发布者发布的消息中需要包含这样的“属性=值”集合,消息将发至与其“属性=值”集合相匹配的队列中去。

1.1.3 路由值(routing key)

在大多数情况下,exchange通过检查一个称为routing key的单个关键值来决定如何路由消息,因而它也被称为一个虚拟地址。对于点对点的路由,routing key就是消息队列的名称;对于主题订阅发布路由,routing key就是主题前缀值。Routing key同样可以基于消息头文件或是文件内容,这都可以自主决定。

1.2 AMQP

AMQP是用来进行应用程序和消息代理服务器之间通信的协议,具有多通道、可协商、异步、安全、简便、中立和高效等特征。它支持金融业的语义机制,提供满足金融业需求的高性能服务支持,支持对新队列和路由技术的扩展,允许应用程序对服务的编程开发。

AMQP的设计需求目标是:

- 采用二进制编码以提高数据的解压包速度。
- 能够处理任意大小的消息而没有特别的限制。
- 在一个连接上能够载有多个通道。
- 对连接的存在时间没有限制。
- 允许异步的命令流水线操作。

AMQP通常可分为三层,如图2所示。



图2 AMQP协议层

模型层定义了一系列代表应用程序执行操作的命令;会话层提供了应用程序和服务器之间具有重播、同步和错误处理的可靠的传输命令;传输层提供框架、多路通道技术、故障检测和数据的传输表示。协议允许自定义传输层而不会影响协议的应用功能,也支持在相同的会话层之上运行不同的高层协议。

2 主动状态证书及其撤销模型

2.1 主动状态证书

主动状态证书是由CA中心或其委托者签发的短周期证

书,用于标志用户证书的当前状态,如图3所示。主动状态证书包括版本号、签名算法、颁布者、证书序列号、主体名称、证书状态、本次发布时间、下次发布时间和签名值等信息。其中,签名算法、证书序列号、主体名称与所标志的证书一致。

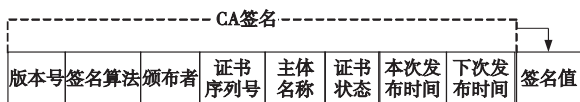


图3 主动状态证书格式

为了支持主动状态证书撤销验证机制,需要在证书中增加主动状态证书发布点等扩展项,来标示如何获取该证书对应的主动状态证书信息。如何定义扩展在文献[4]中有详细描述。

基于主动状态证书的证书撤销机制的基本原理是:CA中心除了为证书用户签发证书外,还同时签发一张描述该证书状态的主动状态证书,并定期更新发布。证书用户根据需要对CA中心订阅所关心的证书状态信息,CA中心根据证书用户的订阅需要及时更新发布所需的状态证书集,证书用户通过获得所订阅的证书状态信息来确定所关心的证书是否被撤销。

主动状态证书撤销机制与周期性证书撤销信息发布机制相比,省去了周期性下载证书撤销列表,极大地减轻了网络的通信负载,而且避免了证书用户从大量不需要的撤销信息中找到所需信息的负担;与在线查询验证机制相比,最大的优点就是提供了异步证书撤销查询的机制,保证了用户在订阅相关证书信息后可以随时上线获取证书撤销信息。同时,通过该机制,在CA中心及时发布已撤销的证书信息的情况下,证书用户可以随时获取最新的证书状态信息,能够满足实时性要求较高的应用需求。

2.2 主动状态证书撤销模型

发布/订阅(pub/sub)是消息中间件的一种工作模式,具有异步以及松耦合的特点^[8]。由于消息的生产者和消费者在时间、空间和控制流等方面完全解耦,使得通信双方可以通过独立地发布、订阅消息进行异步通信,较好地满足了大规模、动态变化的分布式信息系统的集成需要。目前基于主体的发布/订阅已成功应用于金融、股票、移动计算等场合。

从主动状态证书的证书撤销机制原理可以看出,其采用的是一种发布/订阅的模式来实现对状态证书的获取。发布/订阅作为一种消息中间件的工作模式,在AMQP消息中间件中通过配置相关AMQP模型组件就能够方便、快捷地得以实现,因此本文基于AMQP消息中间件提出一种发布/订阅模式的主动状态证书撤销模型。

主动状态证书撤销模型如图4所示。颁发者根据主动证书发布策略发布更新主动状态证书,证书用户订阅所关心证书对应的主动状态证书;AMQP消息中间件将主动状态证书及时发送给正确的证书用户。

3 基于AMQP消息中间件的主动状态证书撤销机制实现

3.1 AMQP消息中间件产品的选择与配置

目前AMQP消息中间件产品的实现有OpenAMQ、Apache Qpid、RabbitMQ等,本文采用支持AMQP的OpenAMQ消息代

理中间件产品。OpenAMQ是由参与设计AMQP的iMatix公司开发的,是AMQP的开源实现,可运行于Linux、AIX、Solaris、Windows、OpenVMS。它具有高度的健壮性和较快的速度,实现了用户所需要的AMQP中设定的功能,在此基础上添加了为方便应用程序开发的称为WireAPI的标准API^[9]和为方便远程管理的称为CML的标准API,并提供了一个将各个服务器连接成整体的标准模型。

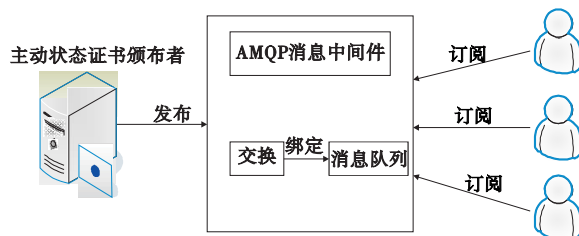


图4 主动状态证书撤销模型

OpenAMQ服务器端启动时应根据需要进行配置,OpenAMQ提供了多种配置方式,本文采用以“amq_server.cfg”命名的XML文件形式进行配置,它能在服务器启动时作用于所有之上的实例。

```
amq_server.cfg:
<? xml version = "1.0" encoding = "UTF-8" ?>
<config>
  <server
    port = "5672"
    listen = " * "
    queue_timeout = "0"
    vhost = "/"
    direct = "1"
  />
  <resources
    monitor = "0"
    dump_state = "60"
  />
  <logging
    debug_route = "0"
    debug_queue = "0"
    debug_peering = "0"
  />
</config>
```

可以看出,该配置文件包括服务器、资源和日志记录三项,其主要是对Server中端口和虚拟主机的配置。端口开启“5672”,虚拟主机采用服务程序启动时的默认项。通过配置,OpenAMQ进入正常的工作状态,同时开启了OpenAMQ提供的direct模式。Direct模式是OpenAMQ为了提高消息的转发性能而设计的称为直接消息协议(DMP)的一种实现方式。由于证书的用户较多,发布端每次发布的消息量会比较大,为了提高效率、降低延迟,本文采用此种可提高消息转发速率达五倍以上的direct模式。

3.2 发布/订阅端的实现

OpenAMQ为发布/订阅端的编程实现提供了WireAPI这一强有力的支持,WireAPI是与OpenAMQ服务器基于同一技术开发的,为客户端提供了针对AMQP的基于组成成分的接口。它的功能是:提供全面的基于异步的操作;能工作于单线程及多线程的应用程序中;提供对AMQP方法属性的全面支

持;能够与任意的基于AMQP的服务器程序工作。

1) 消息格式定义

AMQP中的消息格式如图5所示,该消息格式由五部分组成,分别是消息总长度、交换名、路由值、消息属性和消息体。

size of following message data
[exchange] exchange name (AMQP short string)
[routing key] routing key (AMQP short string)
[property...] header flags + properties (AMQP format)
[body] content body (length + data)

图5 AMQP消息格式

结合AMQP消息格式及提高主动状态证书的发送效率,本文设计了一种消息格式。针对订阅用户对证书的需求是明确的,也为了提高路由的匹配速度,因而采用了直接匹配的方式。消息格式设定为[exchange] type = direct, [routing key] = certificateSerialNumber,属性项置空,消息体为主动状态证书,其中[routing key]中的证书序列号与消息体中的主动状态证书序列号是一致的。

2) 发布/订阅过程

a) Subscriber在OpenAMQ服务器上建立自己的队列,同时将该队列与相应exchange建立相关的订阅binding,使其与exchange绑定。

b) Publisher将消息发布至OpenAMQ服务器中相应类型的exchange,消息则经由该exchange转发至订阅了该消息的队列中。

c) Subscriber可根据需要随时从该队列中通过调用接收函数接口获取自己所订阅的消息。

3) 发布/订阅端的实现

发布/订阅端可统称为客户端。客户端与服务器之间建立发布/订阅系统的大体过程是:客户端与服务器之间经过认证,建立连接和会话;客户端通过调用相关函数接口,在服务器中配置AMQP模块组件,使各组件建立逻辑上的连接关系。

图6所示为客户端程序与服务器程序进行通信的步骤,通过调用WireAPI中的功能函数接口实现服务器对客户端的认证,并建立连接和会话,为之后消息的发布和订阅奠定基础。

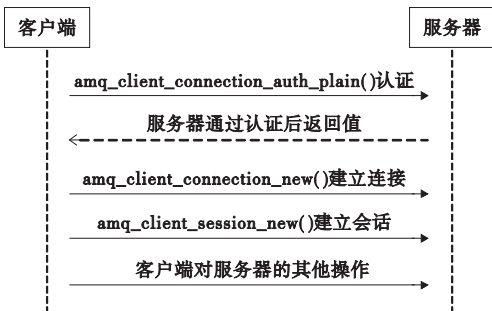


图6 客户端与服务器的通信

建立通信连接的程序如下:

```

//初始化系统
icl_system_initialise (argc, argv);
// 建立连接
auth_data = amq_client_connection_auth_plain ("guest", "guest");
connection = amq_client_connection_new ( host-address, "/", auth_data, "client-name", 0, 30000);
// 开启 direct 模式
if (direct_mode)

```

```

connection->direct = TRUE;
icl_longstr_destroy (&auth_data);
// 建立会话
session = amq_client_session_new (connection);

```

与服务器建立通信之后,就需要对服务器中的AMQP各组件进行配置和逻辑连接。

对发布端程序而言,需要配置exchange组件,生成符合上文结构的AMQP消息,并将消息发布给exchange。

发布端调用的相关函数接口如下:

```

//生成消息
content = amq_content_basic_new ();
amq_content_basic_set_body (content, status-certificate, strlen (status-certificate), free);
//设定 exchange 类型,本文选择 direct 类型
amq_client_session_exchange_declare ( session, 0, amq.direct, "direct", 0, 0, 0, 0, arguments_table);
//发布消息
amq_client_session_basic_publish ( session, content, 0, "amq.direct", CertificateSerialNumber, FALSE, FALSE);

```

对订阅端程序而言,需要配置queue、binding组件,以及从queue中取出订阅的消息。

订阅端调用的相关函数接口如下:

```

// 建立订阅者的私有队列
amq_client_session_queue_declare ( session, 0, subscribe queue, FALSE, TRUE, TRUE, FALSE, NULL);
// 将队列与 exchange 进行绑定
amq_client_session_queue_bind ( session, 0, subscribe queue, "amq.direct", CertificateSerialNumber, NULL);
// 从队列中读取消息
amq_client_session_basic_consume ( session, 0, subscribe queue, NULL, TRUE, TRUE, TRUE, NULL);

```

通过以上配置,建立了如图1所示的AMQP组件exchange、binding和message queue之间的逻辑连接,经过调用发布和从队列中读取消息的函数接口,完成了消息的整个发布/订阅流过程,从而实现了该消息中间件系统。

4 性能测试

为了分析该消息中间件系统的性能,本文依托实验室的局域网构建了一个实验系统。实验环境如下:服务程序采用OpenAMQ-1.4c版本,发布/订阅客户端均采用Microsoft Visual Studio 2008开发,所使用机器均为CPU为Intel Pentium IV 3.0 GHz,内存为512 MB的普通PC机,所使用的操作系统为WindowsXP,连接环境为一小型局域网。

在模拟实验中,分别假设消息以1 k、5 k、10 k的大小发送,同时有100、250、500、750、1 000名订阅者根据不同的需求订阅消息,通过服务器对所有订阅者完成服务的时间来检测服务程序的性能。

图7显示本系统随着同时订阅人数的增加,服务完成时间快速增大,且当订阅人数在250人以上时,服务时间与订阅人数基本成比例增长。

通过实验可表明:a)以10 k的消息量,完成1 000个同时在线订阅服务的时间还不足2 500 ms,表明该订阅/发布系统的速度还是比较快的;b)在本文所给的消息量的范围内,可以看出其对服务的完成时间影响不大,特别是在订阅人数较少的情况下更是不明显,这表明系统可支持用户订(下转第1925页)

(上接第 1921 页)阅相对较大的消息,而不会对系统的性能造成较大的影响;c)在实验环境中系统支持 1 000 人同时在线服务时的时间不足 2 500 ms,表明系统可支持较多的用户订阅服务,在提高服务器性能的条件下,可支持更大的用户数。

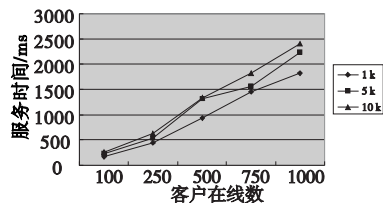


图7 实验结果

5 结束语

采用状态证书的证书撤销机制,可以大幅度地降低证书撤销验证过程中的网络通信负载,有效地减少密码运算次数。本文从证书应用实际出发,基于 AMQP 消息中间件设计了一个支持主动状态证书发布与订阅的系统,实现了状态证书的证书撤销机制。该系统操作简单,支持异步通信,用户可自主订阅服务并随时上线获取消息,且该系统性能稳定、高效。

参考文献:

[1] HOUSLEY R, FORD W, POLK W, *et al.* RFC 2459, Internet X.509 public key infrastructure certificate and CRL profile [S].

1999.

- [2] ÅRNES A, JUST M, KNAPSKOG S J, *et al.* Selecting revocation solutions for PKI[C]//Proc of 5th Nordic Workshop on Secure IT Systems. 2000:233-236.
- [3] MYERS M, ANKNEY R, MALPANI A, *et al.* RFC 2560, X.509 Internet public key infrastructure online certificate status protocol—OCSP[S]. 1999.
- [4] XIONG Xuan-dong, LI Zuo-hui, GAO Xiang. Certificate revocation scheme based on active status certificate[C]//Proc of the 3rd International Conference on Computer and Network Technology. 2011: 365-368.
- [5] Advanced message queuing protocol website[EB/OL]. <http://www.amqp.org/>.
- [6] SUBRAMONI H, MARSH G, NARRAVULA S, *et al.* Design and evaluation of benchmarks for financial applications using advanced message queuing protocol (AMQP) over infiniband[C]//Proc of Workshop on High Performance Computational Finance. 2008:1-8.
- [7] VINOSKI S. Advanced message queuing protocol[J]. *IEEE Internet Computing*, 2006, 10(6):87-89.
- [8] 李璐,张广泉.消息中间件的体系结构研究[J].苏州大学学报:工科版,2007,27(3):10-14.
- [9] Programming WireAPI[EB/OL]. <http://www.openamq.org/>.