ICS 33.040.40
L 78

YD

中华人民共和国通信行业标准

YD/T 2172-2010

IP 网络端点准入控制框架技术要求

Framework technical specification for
IP network on endpoint admission control

2010-12-29 发布　　2011-01-01 实施

中华人民共和国工业和信息化部　发布

目　　次

前　言

本标准的附录A为资料性附录。

本标准由中国通信标准化协会提出并归口。

本标准起草单位：工业和信息化部电信研究院、杭州华三通信技术有限公司、上海贝尔股份有限公司。

本标准主要起草人：杨剑锋、万晓兰、张立新。

IP网络端点准入控制框架技术要求

1 范围

本标准规定了IP网络端点准入控制的框架技术要求，包括对网络接入端点的安全状态评估、安全策略实施、安全策略管理、用户管理、安全联动控制、实时监控和安全状态感知客户端防病毒联动插件接口等方面的要求。

本标准适用于IP网络端点准入控制的开发、测试以及管理。

2 术语、定义和缩略语

2.1 术语和定义

下列术语和定义适用于本标准。

2.1.1

端点

接入网络的用户终端，包括PC、PDA、打印机、IP电话等设备。

2.1.2

安全状态感知点

安全状态感知点在IP网络端点准入控制体系中，端点的位置通常就是安全状态感知点。通过安装在网络端点上的安全客户端，端点的安全信息可被采集并上报。

2.1.3

安全策略执行点

具体实施网络接入授权的网络节点。安全策略决策点根据端点安全状态下发网络访问权限给安全策略执行点，安全策略执行点负责应用端点的接入控制权限。

2.1.4

安全策略决策点

负责评估端点的安全状态。安全策略感知点将终端的硬件、软件、防病毒、系统补丁等与接入安全相关的信息发送给安全策略管理决策点，安全策略决策点根据管理员设置的端点安全策略，对端点的安全状态进行评估，并根据评估结果将网络访问权限下发至安全策略执行点。

2.1.5

安全区

端点的安全认证通过时能访问的网络资源所形成的区域。

2.1.6

隔离区

端点的系统补丁、病毒库版本等安全状态不符合基于用户账号预定义的安全策略时，端点将被限制网络访问权限，只能访问病毒服务器、补丁服务器等用于系统修复的网络资源，这些受限的网络资源被称之为隔离区。

2.2 缩略语

下列缩略语适用于本标准。

ACL	Access Control List	访问控制列表
AV	Anti-virus	防病毒
EAP	Extensible Authentication Protocol	可扩展认证协议
IP	Internet Protocol	互联网协议
PEAP	Protected Extensible Authentication Protocol	受保护的扩展认证协议
VLAN	Virtual Local Area Network	虚拟局域网
VPN	Virtual Private Network	虚拟专用网
WLAN	Wireless Local Area Network	无线局域网
TLS	Transparent LAN Service	透明局域网服务

3 体系结构

3.1 概述

IP网络端点准入控制体系从网络终端入手，将终端防病毒、补丁修复等终端安全措施与网络接入认证、访问权限控制等网络安全措施整合为一个联动的安全体系，通过对网络接入终端的检查、隔离、修复、管理和监控，使整个网络变被动防御为主动防御、变单点防御为全面防御、变分散管理为集中策略管理，提升了网络对病毒、网络攻击等安全威胁的整体防御能力。

3.2 典型组网

IP网络端点准入控制的典型组网结构如图1所示。

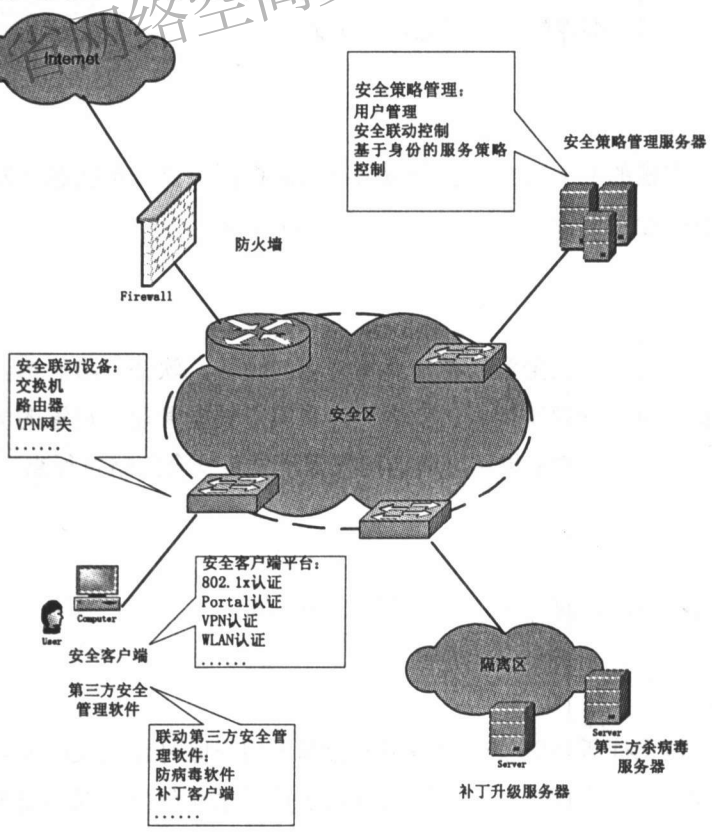

图 1 IP 网络端点准入控制组网结构

图1中端点所连接的接入设备可以是交换机、路由器，也可以是VPN网管等。IP网络端点准入控制的基本原理如下：

1）端点（终端用户）试图接入网络时，首先通过安全客户端进行用户身份认证，非法用户将被拒绝接入网络；

2）合法用户将被要求进行安全状态认证，由安全状态决策点（安全策略服务器）验证端点（用户终端）安全状态是否符合基于用户账号预定义的安全策略，包括补丁版本、病毒库版本是否合格，软件安装允许是否合格、是否使用代理服务器等信息，不合格用户将被安全联动设备隔离到隔离区；

3）进入隔离区的端点只能访问病毒服务器、补丁服务器等用于系统修复的网络资源，也可以卸载非法程序、取消代理设置等操作，直到安全状态合格；进入隔离区的端点无法访问安全区的网络资源。

4）安全状态合格的用户将实施由安全状态决策点下发安全设置，并由安全联动设备提供基于身份的网络服务。

3.3 体系结构

IP网络端点准入控制体系结构如图2所示。

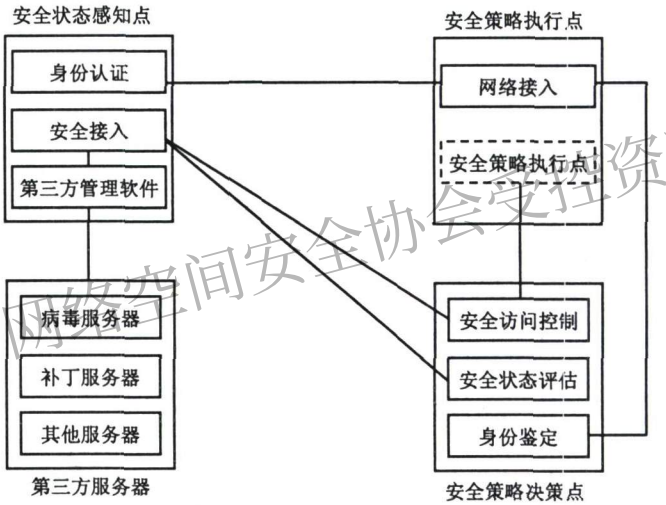

图 2 体系结构

3.3.1 安全状态感知点

安全状态感知点应支持功能如下：

1）安全状态感知客户端与第三方管理软件接口应开放，以便实现联动；

2）安全状态感知客户端应平台化，应支持多种认证插件，宜通过 802.1x、Portal、VPN 等实现；

3）应确保与安全策略决策点的通信安全，信息传输通道应加密。

3.3.2 安全策略执行点

安全策略执行点按构成可分为两种形式：由接入控制设备构成；由客户端构成。

安全策略执行点由交换机、路由器、VPN网关、无线控制器等接入控制设备构成时，应支持的功能如下：

1）实施安全策略；

2）强制用户准入认证；

3）应支持多种认证方式，宜通过 802.1x 认证、Portal 认证、VPN 认证等实现；

4）实施安全隔离（如 ACL、VLAN 控制）；

5）解除隔离状态；

6）实施基于用户的服务策略（如动态 ACL、动态 VLAN）。

安全策略执行点由客户端构成时，应支持的功能如下：

1） 实施安全策略；

2） 需配合交换机、路由器、VPN 网关、无线控制器等进行准入认证；

3） 实施安全隔离（如 ACL 控制）；

4） 解除隔离状态。

3.3.3 安全策略决策点

安全策略决策点应支持下列功能：

1） 安全策略管理；

2） 用户管理；

3） 安全联动控制；

4） 安全状态评估；

5） 安全事件审计；

6） 实时监控。

4 功能组成

4.1 功能流程

IP网络端点准入控制的基本功能是通过安全状态感知客户端、安全联动设备（如交换机、路由器）、安全策略决策点以及第三方安全或桌面管理设备的联动实现，其基本原理如图3所示。

图 3 IP 网络端点准入控制功能流程

4.2 认证功能

IP网络端点准入控制应支持多种认证方式，通过不同方式的身份验证技术，以确保接入终端的身份合法。

IP网络端点准入控制的身份认证宜通过802.1x、Portal、VPN等实现。根据应用场景的区别，选择合适的方式实施准入防御策略，分别从局域网接入、VPN接入、WLAN接入等不同层面确保网络的整体安全。

4.3 安全状态评估功能

4.3.1 用户终端安全状态检查

IP网络端点准入控制应支持用户终端的安全状态检查功能。

IP网络端点准入控制应支持通过对终端安全状态的检查，使得只有符合特定的安全标准的终端才能正常访问网络，以便于提高网络的安全性和稳定性。

用户终端安全状态检查内容包括：

1）操作系统补丁、防病毒软件版本病毒库版本、是否感染病毒等反映终端防御能力的状态信息；

2）用户终端安装的软件、启动的服务是否符合特定的网络安全需求；

3）双网卡使用限制、IE 代理使用、ARP 欺骗攻击等网络终端安全防御要求。

4.3.2 检查报告发送

IP网络端点准入控制应支持发送检查报告功能。

安全状态感知客户端与第三方安全管理软件联动检查终端安全状态后，应能将检查报告发送给安全策略决策点，由安全策略决策点根据特定的安全标准检查终端安全状态是否合格。

4.4 隔离功能

IP网络端点准入控制应支持隔离“危险”和“易感”终端的功能。

IP网络端点准入控制要求系统补丁、病毒库版本不及时更新或已感染病毒的用户终端，应限制访问权限，只能访问病毒服务器、补丁服务器等其他第三方安全管理用于系统修复的网络资源，这些受限的网络资源被称之为“隔离区”，通过ACL或VLAN方式实现。

4.5 修复功能

IP网络端点准入控制应支持修复功能。

IP网络端点准入控制可通过强制的方式修复系统补丁、升级防病毒软件。IP网络端点准入控制使用强制修复方式时，不符合安全策略的用户终端应被限制到“隔离区”，并且自动提醒用户进行补丁更新或最新病毒库的升级，并联动防病毒服务器、补丁服务器或其他第三方安全管理软件，帮助用户完成手工或自动升级操作，达到提升终端主动防御能力的目的。完成修复并达到安全策略的要求后，用户终端将被取消隔离，可以正常访问网络。

4.6 监控与管理功能

IP网络端点准入控制应支持监控与管理功能。

集中、统一的安全策略管理和安全事件监控是IP网络端点准入控制的重要功能。特定安全策略的统一实施，需要有一个完善的安全策略管理平台来支撑。IP网络端点准入控制应支持接入策略、安全策略、服务策略、安全事件监控统一于一体的用户管理平台，帮助网络管理员制定基于用户身份的、个性化的网络安全策略；同时IP网络端点准入控制应支持通过安全策略决策点与安全状态感知客户端

的配合，强制终端实施安全配置（如是否实时检查邮件、注册表、是否限制代理、是否限制双网卡等），监控用户终端的安全事件（如查杀病毒、修改安全设置等）。

5 功能要求

5.1 认证

5.1.1 身份认证

IP网络端点准入控制应支持多种认证方式。身份认证可通过802.1x、Portal、VPN方式等实现。特定的网络设备宜根据实际应用情况和场景，选择采用有效的认证方式，如：

1）802.1x 认证与交换机、BAS 设备在接入层、汇聚层实现对接入用户的端点准入控制；

2）Portal 认证与路由器或者具有路由功能的交换机在网络出口实现对接入用户的准入控制；

3）VPN 认证与 VPN 网关的配合实现对远程接入用户的端点准入控制。

用户终端接入网络时，应首先通过安全状态感知客户端进行用户身份认证，非法用户将被拒绝接入网络。

身份认证宜支持：

1）用户特征绑定，通过用户账号和用户特征（用户特征即 VLANID、PORT、MAC、IP，这些用户特征可以任意组合都可以于用户账号绑定）绑定来增强用户的安全性；

2）动态用户绑定，实现对用户 IP 和 MAC 的动态绑定，当用户上线后，就不允许用户再更改 IP 地址，以防止用户账户被盗用；

3）支持 EAP TLS、PEAP 等认证方式。

5.1.2 安全状态评估

IP网络端点准入控制应支持安全状态评估功能。

用户终端在身份认证之后，安全状态感知客户端与补丁客户端、防病毒客户端、第三方安全管理软件联动，检查用户终端的安全状态，包括操作系统版本、系统补丁、用户终端的防病毒软件版本、病毒库版本、以及防火墙等信息。

用户终端的安全信息将被传递到安全策略决策点，如果符合安全策略决策点上特定的安全标准，则安全评估通过；否则，不通过。

5.2 安全策略实施

5.2.1 执行安全策略

IP网络端点准入控制应支持执行安全策略的功能。

安全状态感知客户端接收安全策略决策点下发的安全策略并强制用户终端执行，包括：

1）设置安全策略（是否监控邮件、注册表等）；

2）系统修复通知与实施（自动或手工升级补丁和病毒库等）功能。

5.2.2 用户准入控制

IP网络端点准入控制应支持用户准入控制功能，包括：

1）强化用户身份认证；

2）基于身份的服务策略下发（动态 ACL、动态 VLAN 等）；

3）不符合安全策略的用户终端将被限制在隔离区。

5.3 安全策略管理

IP网络端点准入控制应支持安全策略管理功能。

安全策略决策点定义了用户终端准入控制的一系列策略，宜包括：

1）客户端安全状态评估配置；

2）防病毒、升级信息提示配置；

3）补丁检查项配置；

4）安全策略配置（是否实时检查邮件、注册表，是否限制代理，是否限制双网卡）；

5）安全设置检查（是否设置 IE 代理，是否启用双网卡、屏保时长、查杀病毒级别）；

6）隔离区（ACL、VLAN）设置；

7）服务策略配置（接入时段、ACL、VLAN）。

5.4 用户管理

IP网络端点准入控制应支持用户管理功能。

网络中不同的用户、不同类型的接入终端可能要求不同级别的安全检查和控制，安全策略决策点应为不同用户提供基于身份的个性化安全配置和网络服务等级，宜包括：

1）用户身份验证策略配置；

2）用户级安全策略设置；

3）基于身份的网络服务策略配置；

4）用户安全状态评估结果审计；

5）用户安全事件查询，掌握网络安全状态；

6）用户上网记录查询。

5.5 安全联动控制

IP网络端点准入控制应支持安全联动功能。

安全策略决策点负责评估安全状态感知客户端上报的安全状态，下发用户终端的修复方式与安全策略来控制安全策略执行设备对用户的隔离与开放。在安全策略决策点的控制下，安全状态感知客户端、安全策略执行点与防病毒服务器、补丁服务器或其他第三方安全管理软件协同工作，配合完成端到端的安全准入控制。联动控制宜包括：

1）用户安全状态检测策略下发；

2）用户安全状态评估（审核）；

3）用户安全策略下发（是否实时检查注册表、是否限制代理、是否限制双网卡）；

4）安全事件接收与处理；

5）用户隔离控制；

6）用户安全日志收集。

5.6 实时监控

安全状态感知客户端应对用户终端进行实时监控：

1）是否更改安全设置；

2）是否更改网络配置信息；

3）是否发现新病毒；

4）是否安装和使用非法软件；

5）用户安全日志审计。

6 接口要求

6.1 安全状态感知接口要求

安全状态感知接口是安全状态感知客户端与安全联动软件之间的接口。

安全状态感知客户端允许通过接口调用获取安全联动软件的状态信息，执行安全策略决策点下发的安全操作。

安全状态感知接口包括安全插件注册接口、安全状态查询接口和安全任务执行接口三个子功能接口。

1）安全插件注册接口：实现对安全软件的注册。安全状态感知客户端可以从注册数据库查询已安装的安全联动软件。比如防病毒软件、防火墙软件或补丁联动软件。

2）安全状态查询接口：实现对各种安全信息的查询，如防火墙软件提供防火墙规则的查询接口。安全状态感知客户端可以通过该接口获取防火墙规则并上报安全策略决策点。

3）安全任务执行接口：负责执行安全状态感知客户端发起的安全任务，如全盘扫描病毒。

6.2 安全状态控制接口要求

安全状态控制接口是安全状态感知客户端与安全策略决策点之间的接口。

安全状态感知客户端需要向安全策略决策点提供端点的安全状态，同时，安全策略决策点也要向安全状态感知点下发安全控制指令。

安全状态感知点和安全策略决策点之间的信息交互宜使用安全状态控制协议实现。安全状态控制协议要求待定。

6.3 安全状态实施接口要求

安全状态实施接口是安全策略执行点与安全策略决策点之间的接口。

安全策略决策点的网络接入的阻断、隔离等信息，需通过安全状态实施接口下发给安全策略执行点。IP网络端点准入控制的体系架构，是基于用户的身份认证而实现的。安全策略执行点与安全策略决策点之间的信息交互建议使用扩展RADIUS协议，实现IP端点准入的控制。

附 录 A
（资料性附录）
安全状态感知客户端防病毒联动插件接口的实现

A.1 安全状态感知接口

安全状态感知客户端与防病毒联动插件之间的接口为安全状态感知接口。安全状态感知客户端防病毒联动插件可由第三方厂商提供，该插件仅由安全状态感知客户端使用，为单向接口，第三方客户端软件无法通过该插件获取安全状态感知客户端的状态，或向安全状态感知客户端提交任务执行请求。

安全状态感知客户端防病毒联动插件分为两部分：动态链接库文件（如av-plugin.dll）和插件配置文件（如av-plugin.inf）。动态链接库文件用于提供接口调用，配置文件用于指明插件的相关信息。

安全状态感知客户端防病毒联动插件的作用是安全状态感知客户端对第三方客户端软件进行状态查询（Posture Query）及提交任务执行请求（Task Invoke Request）。

A.1.1 状态查询

安全状态感知客户端通过调用安全状态感知客户端联动插件，可查询第三方客户端的执行状态信息。

A.1.2 任务执行请求

安全状态感知客户端通过任务请求，向第三方客户端软件分配任务。若需要执行结果，通过异步方式等待执行结果。

A.2 安全状态感知客户端防病毒联动插件的安装、升级和装载

A.2.1 联动插件安装、升级和装载涉及的相关目录

安全状态感知客户端防病毒联动插件安装、升级和装载涉及的目录有：

插件安装目录，例如：%CommonProgramFiles%\CompanyName\Secure Client\Plugins\<SOME PLUGIN>\Install；

插件装载目录，例如：%CommonProgramFiles%\CompanyName\Secure Client\Plugins\<SOME PLUGIN>；

插件隔离区，例如：%CommonProgramFiles%\CompanyName\Secure Client\Plugins\<SOME PLUGIN>\Quarantine。

上述示例中，“%CommonProgramFiles%”是指Windows系统环境变量所指定的路径，一般为Windows安装逻辑分区下的“\Program Files\Common Files”子目录。当Windows安装在C:，则该环境变量的值为“C:\Program Files\Common Files”。

“<SOME PLUGIN>”用于唯一标识一个插件，命名格式为“<插件提供商缩写>_<插件类型>”。如：CCIA_Plugin_AV，这里的“CCIA”是假设的某个插件提供商的缩写，“Plugin_AV”是针对防病毒插件的插件类型。

插件的安装与升级由第三方软件产品提供，随第三方客户端软件的安装与升级，实现插件的安装与升级，而不是单独为插件提供安装程序。

A.2.2 插件安装说明

在第三方客户端软件安装过程中，自动安装该插件，完成安装后插件对应的两个文件存放在安装目录下。

A.2.3 插件升级说明

在第三方客户端软件升级过程中，自动升级该插件，完成升级后新插件对应的两个文件仍应存放在同一安装目录下。

A.2.4 插件装载说明

插件完成安装或升级后，并不能为安全状态感知客户端使用，还需要完成装载操作，插件完成装载后才能为安全状态感知客户端使用。所谓装载，就是把插件文件从插件安装目录下复制到插件装载目录下。插件装载操作由安全状态感知客户端检查实施。

区分装载与安装/升级，是为了更好地协调第三方客户端软件与安全状态感知客户端的合作分工。第三方客户端软件的安装/升级程序只负责安装/更新插件文件，而插件的装载和启用交给安全状态感知客户端去处理。

插件装载操作的步骤：

安全状态感知客户端每次启动时，检查插件安装目录，如果发现该目录下存在新的插件文件，则执行以下操作：

如果在插件装载目录找不到对应文件，则安全状态感知客户端执行插件装载操作，将插件文件从插件安装目录复制到插件装载目录下。

如果插件装载目录下存在该文件，则安全状态感知客户端先检查待装载的插件是否比原插件新：如果是，则重新装载该插件，即将新的插件文件复制到装载目录下。

如果插件在装载过程中出现失败，则将插件文件复制到插件隔离区。如果需要的话，可以记录日志或向安全管理中心汇报。

插件装载操作要求对待装载的插件文件进行有效性检查，即保证DLL文件与配置文件是匹配的。

DLL文件本身应该含有文件版本、产品名称等版本属性，其值与插件配置文件中对应属性的值要求一致。

插件装载还有一个重要操作：把插件配置文件中的状态查询功能、执行任务功能、错误号及其描述信息更新到安全状态感知客户端使用的插件配置文件中。

A.3 插件调用接口

插件调用接口主要包括两个函数：状态查询接口函数和任务执行请求接口函数。另外，还有两个辅助函数：资源初始化函数和资源释放函数。

A.3.1 状态查询接口函数

A.3.1.1 接口函数原型

```
DWORD       SPI_GetProperty(        //SPI：Security Plug-in Interface
LPCTSTR     lpczPropertyName,       // [in]查询属性名，不可为空
LPBYTE      lpBuffer,               // [out]存放查询结果的缓存
DWORD&      dwSize                  // [in out]查询结果缓存的长度（字节数）
);
```

A.3.1.2 函数参数说明

1）参数1（lpczPropertyName）为输入参数，用于指定查询属性名，即指定本接口函数去查询什么（如："SysQuery:Running-Task"、"AVQuery:Product-Name"等，具体支持的状态查询功能请参考本标准后面两章内容）。

2）参数 2（lpBuffer）为输出参数，用于指定存放查询操作结果的缓存，查询结果统一以如下的XML 格式表示：

```
<queryOutput operation="operation name">
<simpleResult>result</simpleResult>
<complexResult>
<patch>
<item type="add">
<value>patch1</value>
<value>patch2</value>
</item>
<item type="need">
<value>patch3</value>
<value>patch4</value>
</item>
</patch>
<policy>
<item type="enabled">
<value>file</value>
<value>email</value>
</item>
<item type="disabled">
<value>register</value>
</item>
</policy>
<virus>
<item name="virus1" time="date time1" type="memory" result="c"/>
<item name="virus1" time="date time2" type="file" result="i"/>
</virus>
</complexResult>
</queryOutput>
```

约束说明：在queryOutput标签下，simpleResult标签和complexResult标签是互斥的，只能包含其中一个；在complexResult标签下，patch标签、policy标签和virus标签是互斥的，只能包含其中一个。各标签的含义在后面描述。

3）参数 3（dwSize）为输入输出参数，输入时指定存放查询结果缓存的大小；输出时有三种情况：

a）如果函数成功返回，输出时该参数指定查询结果的实际长度（有些查询操作不需要通过参数 2 携带查询结果，此时本参数值为 0）；

b）如果函数返回缓存空间不够对应的错误号（在下面描述），输出时该参数指定查询结果的实际长度；

c）如果函数返回其他错误号，输出时该参数保持输入时的值。

A.3.1.3 函数返回值说明

返回 0 表示操作成功；返回其他值（即错误号）表示操作失败。

本接口可能返回的错误号有：

1）未知属性，即该属性未定义；

2）缓存空间不够，不足于存放查询结果；

3）当调用该函数查询任务执行结果时，返回该值表示任务正在执行过程中；

4）插件对应的应用程序异常；

5）当前没有执行任务；

6）任务执行失败；

7）任务被强行终止。

针对特定的插件，本函数可以定制新的错误号。约定插件自己定制的错误号应大于 10000。而[1，10000]内的错误号为所有插件公用，称为公用错误号。

公用错误号对应的错误描述信息在安全状态感知客户端的公用配置文件中记录，而插件定制的错误号及对应的错误信息则在插件的配置文件中记录。

A.3.1.4 其他说明

该函数在获得查询结果前，不应返回。

A.3.2 任务执行请求接口函数

A.3.2.1 接口函数原型

```
DWORD SPI_ExecuteTask(
    LPCTSTR            lpczTaskName,      // [in] 任务名称
PPARAMETERS_INFO  pParameters,       // [in] 任务参数
);
```

A.3.2.2 函数参数说明

1）参数 1（lpczTaskName）为输入参数，用于指定待执行的任务名称（如："SysTask:Task-Terminate"、"AVTask:Kill"等，具体支持的执行任务功能请参考本标准后面章节的内容）。

2）参数 2（pParameters）为输入参数，用于指定任务执行所需的参数。

A.3.2.3 函数返回值

0表示操作成功；

返回其他值（即错误号）表示操作失败。

本接口可能返回的错误号有：

101：指定的任务不支持/不存在；

102：任务参数非法；

103：任务提交失败。

针对特定的插件，本函数可以定制新的错误号。约定插件自己定制的错误号应大于10000。而[1，10000]内的错误号为所有插件公用，称为公用错误号。

公用错误号对应的错误描述信息在安全状态感知客户端的公用配置文件中记录，而插件定制的错误号及对应的错误信息则在插件的配置文件中记录。

A.3.2.4 其他说明

该函数调用后，应立即返回，表示任务提交成功。也就是说，本接口函数只负责向防病毒客户端提交指定的任务，而任务的具体执行则交给防病毒客户端去做。

安全状态感知客户端调用本接口函数成功提交一个任务后，将以特定时长为周期，轮询调用SPI_GetProperty("SysQuery:Task-Result",…)以获取任务执行的结果。在任务执行完成并得到结果之前，该查询操作总是返回错误号3（任务正在执行）。

任务参数的结构定义为：

```
typedef struct _PARAMETERS_INFO {
    DWORD       dwParametersCount;      // 参数个数
    PPARAMETER  pfirstParameter;        // 参数链头
} PARAMETERS_INFO, *PPARAMETERS;

typedef struct _PARAMETER_INFO {
DWORD       dwType;         // 参数类型（1——整型；2——字符串；3——任意二进制数据）
LPCTSTR     lpczName        // 参数名称
LPBYTE      lpbData;        // 参数值
DWORD       dwSize;         //参数值的长度（当参数类型为1时，强制认为参数值的长度为4，而忽略本分量取值）
    PPARAMETER next;                // 指向下一个参数的指针
} PARAMETER_INFO, *PPARAMETER;
```

安全状态感知客户端不允许在同一时刻调用一个插件执行多个任务。即当一个插件正在执行某项任务的过程中，不允许其执行该插件提供的其他任务。系统定义的、用于强制终止任务执行的系统任务（即后面描述的SysTask:Task-Terminate）除外。此特性由安全状态感知客户端保证。

防病毒程序插件只有在接收到新的调用任务请求时，才清除上次任务的执行结果。

A.3.3 资源初始化函数

A.3.3.1 接口函数原型

```
DWORD SPI_Init( );
```

A.3.3.2 函数返回值

0 表示操作成功；

返回其他值（即错误号）表示操作失败。

本接口可能返回的错误号有：

4：插件对应的应用程序异常。

针对特定的插件，本函数可以定制新的错误号。约定插件自己定制的错误号应大于 10000，插件定制的错误号及对应的错误信息则在插件的配置文件中记录（参考后面章节的内容）。

A.3.3.3 其他说明

资源初始化函数仅在iNode启动时，加载完插件DLL之后被调用。

如果插件支持自动启动AV客户端的功能，可以在此隐含使用。

A.3.4 资源释放函数

A.3.4.1 接口函数原型

DWORD SPI_Release( );

A.3.4.2 函数返回值

0表示操作成功；

返回其他值（即错误号）表示操作失败。

本接口可能返回的错误号有：

4：插件对应的应用程序异常。

针对特定的插件，本函数可以定制新的错误号。约定插件自己定制的错误号应大于10000，插件定制的错误号及对应的错误信息则在插件的配置文件中记录（参考后面章节的内容）。

A.3.4.3 其他说明

资源释放函数仅在iNode退出时，在释放插件DLL之前被调用。

A.4 通用状态查询与任务执行功能

通用状态查询与任务执行功能是指所有插件必须实现的状态查询和任务执行功能。

A.4.1 通用状态查询功能

每个通用状态查询功能对应的属性名称格式为“SysQuery:”+相应查询功能的英文描述。

所有插件必须实现以下状态查询功能。

A.4.1.1 SysQuery:Running-Task

1）作用

检查当前正在执行任务的名称。

2）返回值

0——操作成功；

2——缓存空间不够；

4——插件对应的应用程序异常；

5——当前没有执行任务。

3）查询结果

操作成功时为正在执行任务的名称。格式如下：

<queryOutput operation="SysQuery:Running-Task">

<simpleResult>result</simpleResult>

</queryOutput>

4）说明

此处operation的值就是本查询功能对应的属性名，而result处填置为正在执行任务的名称。queryOutput标签下只能包含一个simpleResult标签。

A.4.1.2 SysQuery:Task-Result

1）作用

查询任务执行结果。

2）返回值

0——操作成功；

2——缓存空间不够；

3——任务正在执行；

4——插件对应的应用程序异常；

5——当前没有执行任务；

6——任务执行失败；

7——任务被强行终止。

3）查询结果

操作成功时为任务执行的结果信息（具体结果随任务的不同而不同）；操作失败时为任务执行失败的具体描述信息。

A.4.2 通用任务执行功能

每个通用状态查询功能对应的任务名称格式为：“SysTask:”+相应执行任务的英文描述。

所有插件应实现以下任务执行功能：

SysTask:Task-Terminate

1）作用

强制终止任务执行。

2）参数

无。

3）返回值

0——操作成功；

5——当前没有执行任务；

4——插件对应的应用程序异常；

103——任务提交失败。

4）说明

◆ 返回值为0时，安全状态感知客户端需要周期调用查询接口函数SPI_GetProperty(“SysQuery:Task-Result”,…)，并轮巡到该函数返回错误号7（任务被强行终止）就说明本次任务执行完成，即待强行终止的任务已经被强行终止了。

◆ 当返回值为5时，说明待强行终止的任务已经结束，安全状态感知客户端无需周期调用查询接口函数轮巡查看结果。

A.5 防病毒软件需要实现的状态查询与任务执行功能

A.5.1 防病毒软件需要实现的状态查询功能

每个防病毒软件需实现的状态查询功能对应的属性名称格式为：“**AVQuery**:”+相应查询功能的英文描述。

A.5.1.1 AVQuery:Product-Name

1）作用

查询AV客户端软件名称。

2）返回值

0——操作成功；

1——属性未定义；

2——缓存空间不够。

3）查询结果

操作成功时为防病毒客户端软件名称信息，格式如下：

<queryOutput operation="AVQuery:Product-Name">

<simpleResult>result</simpleResult>

</queryOutput>

4）说明

此处operation的值就是本查询功能对应的属性名，而result处填置为AV客户端软件名称。queryOutput标签下只能包含一个simpleResult标签。

A.5.1.2 AVQuery:Product-Version

1）作用

查询AV客户端软件版本。

2）返回值

0——操作成功；

1——属性未定义；

2——缓存空间不够。

3）查询结果

操作成功时为防病毒客户端软件版本信息，格式如下：

<queryOutput operation="AVQuery:Product-Version">

<simpleResult>result</simpleResult>

</queryOutput>

4）说明

此处operation的值就是本查询功能对应的属性名，而result处填置为AV客户端软件版本号。queryOutput标签下只能包含一个simpleResult标签。

A.5.1.3 AVQuery:Virus-Def-Version

1）作用

查询病毒库定义版本。

2）返回值

0——操作成功；

1——属性未定义；

2——缓存空间不够。

3）查询结果

操作成功时为病毒库定义版本信息，格式如下：

<queryOutput operation="AVQuery:Virus-Def-Version">

<simpleResult>result</simpleResult>

</queryOutput>

4）说明

此处operation的值就是本查询功能对应的属性名，而result处填置为病毒库定义日期（格式为：yyyy-mm-dd）。queryOutput标签下只能包含一个simpleResult标签。

A.5.1.4 AVQuery:Engine-Version

1）作用

查询杀毒引擎版本。

2）返回值

0——操作成功；

1——属性未定义；

2——缓存空间不够。

3）查询结果

操作成功时为杀毒引擎版本号，格式如下：

<queryOutput operation="AVQuery:Engine-Version">

<simpleResult>result</simpleResult>

</queryOutput>

4）说明

此处operation的值就是本查询功能对应的属性名，而result处填置为杀毒引擎版本号，且为日期格式（yyyy-mm-dd）。queryOutput标签下只能包含一个simpleResult标签。

A.5.1.5 AVQuery:IsAvailable

1）作用

查询AV客户端的运行状态。安全状态感知客户端在启动后与AV客户端通信前，首先会通过调用查询接口函数SPI_GetProperty("AVQuery:IsAvailable",...)获取AV客户端的运行状态。只有函数返回0才表示AV客户端的运行状态正常；返回其他值时，安全状态感知客户端会将该返回值对应的错误描述信息写入日志文件（即安全状态感知客户端因何原因退出），然后退出。

2）返回值

0——操作成功，表示AV客户端运行状态正常，所谓正常就是可以响应iNode的查询命令或任务执行请求；

1——属性未定义，表示AV客户端不支持对应的查询功能。只可能是传入SPI_GetProperty()的第一个参数为拼写错误时（如："AVQuery:IsAailable"）才会返回本错误号；

4——插件对应的应用程序异常，即AV客户端没有运行，或者已经运行但是无法响应安全状态感知客户端的查询命令或任务执行请求。如果插件支持自动启动AV客户端的功能，可以在此隐含使用，即插件接到AVQuery:IsAvailable查询命令后，检查到AV客户端没有运行便自动启动AV客户端，启动成功给iNode返回0，启动失败才给iNode返回本错误号。

10001——AV客户端正在杀毒

10002——AV客户端正在升级

特别注意：这两个错误号只是用于插件定制错误号示例（大于10,000），本意是错误号4的细化，并不要求插件实现时一定要有这样的错误号。而且，如果AV客户端正在杀毒的同时还能响应iNode的查询命令或任务执行请求，就应该给iNode返回0，设置错误号10001（AV客户端正在杀毒）就毫无意义。

3）查询结果

无。查询结果是指由SPI_GetProperty的第二个参数带回的XML格式的数据，AVQuery:IsAvailable这个状态查询功能的结果通过返回值来体现，无需通过SPI_GetProperty的第二个参数带回数据，因此描述为无。

A.5.1.6 AVQuery:Security-Strategies

1）作用

查询当前的安全配置策略。

2）返回值

0——操作成功；

1——属性未定义；

2——缓存空间不够。

3）查询结果

当操作成功时为当前的安全配置策略，格式如下：

```
<queryOutput operation="AVQuery:Security-Strategies">
<complexResult>
<policy>
<item type="enabled">
<value>file</value>
<value>email</value>
</item>
<item type="disabled">
<value>register</value>
</item>
</policy>
</complexResult>
</queryOutput>
```

4）说明

queryOutput标签的operation属性的值就是本查询功能对应的属性名，queryOutput标签下包含一个complexResult标签，complexResult标签下包含一个policy标签，一个policy标签下包含两个item标签（type分别为enabled和disabled），一个item标签下可以包含0个、1个或多个value标签。value的取值目前支持的有（根据AV客户端提供的具体功能可以扩展）：

file，对传入电脑的文件进行（不进行）监控；

email，对接收的所有电子邮件进行（不进行）监控；

register，对注册表的改动进行（不进行）监控；

page，对上网浏览的网页进行（不进行）监控；

script，对网站或网页上的恶意脚本进行（不进行）监控；

chat，对QQ等即时通讯信息进行（不进行）监控。

A.5.1.7 AVQuery:Monitor-Info

1）作用

查询实时监控收集到的告警信息。

2）返回值

0——操作成功；

1——属性未定义；

2——缓存空间不够。

3）查询结果

当操作成功时为当前15分钟以内实时监控收集到的告警信息，格式如下：

<queryOutput operation="AVQuery:Monitor-Info">

<complexResult>

<virus>

<item name="virus1" time="time1" type="memory" result="c"/>

<item name="virus2" time=" time2" type="file" result="i"/>

</virus>

</complexResult>

</queryOutput>

4）说明

此处queryOutput标签的operation属性的值就是本查询功能对应的属性名，queryOutput标签下包含一个complexResult标签，complexResult标签下包含一个virus标签，一个virus标签下可以包含0个、1个或多个item标签。一个item标签对应一条告警信息，具体有以下4个属性组成：

（1）name发现的病毒名称。

（2）time发现的日期时间，格式为：yyyy-mm-dd hh:mi:ss，如："2004-10-08 17:51:06"。

（3）type感染目标类型，目前支持的类型有（可根据实际需要扩展）：

a. "memory"内存；

b. "boot"引导扇区；

c. "file"文件；

d. "email"邮件。

（4）result处理结果，目前有（可根据实际需要扩展）：

"c"：clean，从感染目标中清除了感染的病毒；

"i"：isolated，感染目标被隔离；

"d"：deleted，感染目标被删除；

"u"：unclean，不能清除该病毒。

符合安全状态感知客户端需求的AVQuery:Monitor-info查询请求必需满足以下约束：

（1）为保证同一条告警信息不会多次反馈给安全状态感知客户端，AV客户端需要维护一个监控起始时间点（记为：StartTime，可以将其初始化为AV客户端启动后的时间点）。每当收到一个AVQuery:Monitor-info查询请求，需要根据当前StartTime的取值和当时时间点（记为：EndTime）计算出本次反馈的时间段范围，仅限落在该时间段范围内的告警信息可以反馈给安全状态感知客户端。本次反馈的时间段记为：

RealStartTime, EndTime]，

其中，RealStartTime=max{StartTime,EndTime－15分钟}。AV客户端处理完该AVQuery:Monitor-info请求后，要更新监控起始时间点，即把EndTime的值赋给StartTime。

（2）反馈给安全状态感知客户端的任意两条告警信息不重复。两条告警信息重复是指对应的4个属性（name,time,type,result）的值都相等。

（3）如果本次反馈的时间段内没有任何告警信息，则不携带查询结果（或者说查询结果为空），即SPI_GetProperty的第三个参数（DWORD& dwSize）置为0。

（4）如果本次反馈的时间段内的告警信息大于50条，则仅反馈时间点最新的50条给安全状态感知客户端。

A.5.2 防病毒软件需实现的任务执行功能

每个防病毒软件需实现的任务执行功能对应的任务名称格式为：“AVTask:”+相应执行任务的英文描述。

A.5.2.1 AVTask:Security-Strategies-Update

1）作用

提交更新杀毒软件安全策略任务。

2）参数

包括所有需要下发的安全策略属性及其取值，见表A.1。

表 A.1 AVTask:Security-Strategies-Update 任务参数表

	参数值类型	参数名称	参数值
参数类型 1	2——字符串	EnableStrategy	<file> | <email> | <register> | <page> | <script> | <chat> | <all>
参数类型 2	2——字符串	DisableStrategy	<file> | <email> | <register> | <page> | <script> | <chat> | <all>

3）说明

（1） 参数EnableStrategy的取值为<file> | <email> | <register> | <page> | <script> | <chat> | <all>中的一个，如：EnableStrategy=“email”表示要设置AV客户端对接收的所有电子邮件进行监控；而EnableStrategy =“all”，表示设置AV客户端对所有监控项进行监控。

（2） 参数DisableStrategy的取值为<file> | <email> | <register> | <page> | <script> | <chat> | <all>中的一个，如：DisableStrategy =“email”表示要设置AV客户端对接收的所有电子邮件不进行监控；而DisableStrategy =“all”，表示设置AV客户端对所有监控项都不进行监控。

（3） 实际调用任务执行接口函数时，参数EnableStrategy和DisableStrategy都可以使用多次。举例如下：

调用SPI_ExecuteTask()，传入的第一个参数为“AVTask:Security-Strategies-Update”，第二个参数则指向图A.1所示的结构。

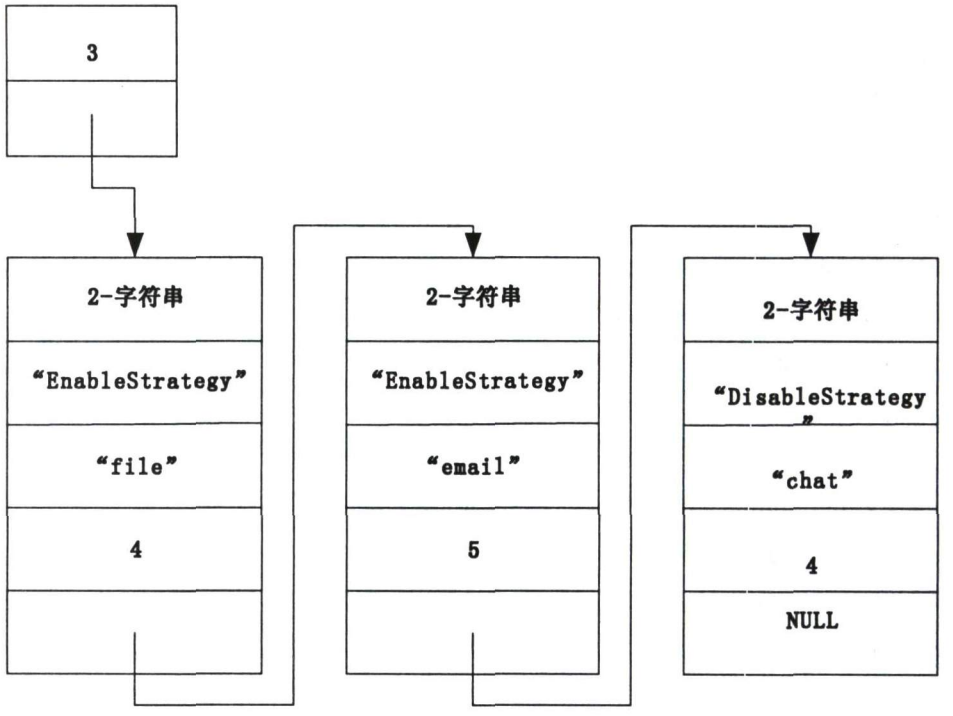

图 A.1 AVTask:Security-Strategies-Update 任务执行功能参数传递示意

4）返回值

0-操作成功；

101-指定的任务不支持/不存在；

102-任务参数非法。

5）说明

当返回值为0时，安全状态感知客户端需要周期调用查询接口函数SPI_GetProperty(“SysQuery:Task-Result”,…)，并轮巡到该函数成功返回0就说明本次任务执行完成。

A.5.2.2 AVTask:Kill

1）作用

提交杀毒任务。

2）参数

见表A.2。

表 A.2 AVTask:Kill 任务参数

	参数值类型	参数名称	参数值
参数类型 1	2——字符串	KillType	<memory> | <disk:all> | <disk:system> | <disk:system-folder> | <file://X:\yyy> | <all>

3）说明

（1） 参数KillType的取值为<memory> | <disk:all> | <disk:system> | <disk:system-folder> | <file://X:\yyy> | <all>中的一个，分别表示要求对内存、全盘、OS安装的所在分区、系统文件夹、指定路径以及内存和全盘进行杀毒任务。

（2） 实际调用任务执行接口函数时，参数KillType可以使用多次。

4）返回值

0——操作成功；

101——指定的任务不支持/不存在；

102——任务参数非法。

当返回值为0时，安全状态感知客户端需要周期调用查询接口函数SPI_GetProperty(“SysQuery:Task-Result”,…)，并轮巡到该函数成功返回0就说明本次任务执行完成，并通过SPI_GetProperty的第二个参数带回任务的执行结果。执行结果格式如下：

<queryOutput operation=”AVTask:Kill”>

<complexResult>

<virus>

<item name=”virus1” time=”time1” type=”memory” result=”c”/>

<item name=”virus2” time=” time2” type=”file” result=”i”/>

</virus>

</complexResult>

</queryOutput>

queryOutput标签的operation属性的值就是本次任务对应的任务名称，除此之外，本次任务执行结果与AVQuery:Monitor-Info查询功能的查询结果在格式上是一样的。

任务执行结果满足以下约束：

（1）反馈给iNode的任意两条病毒信息不重复。两条病毒信息重复是指对应的4个属性（name,time,type,result）的值都相等。

（2）如果本次任务没有发现任何病毒，则不携带查询结果（或者说查询结果为空），即SPI_GetProperty(“SysQuery:Task-Result”,...)的第三个参数（DWORD& dwSize）置为0。

（3）如果本次任务发现的病毒条目大于50条，则仅反馈时间点最新的50条给安全状态感知客户端。

A.5.3 AVTask:Virus-Def-And-Engine-Update

1）作用

提交升级病毒库以及杀毒引擎任务。

2）参数

无。

3）返回值

0——操作成功；

101——指定的任务不支持/不存在。

当返回值为0时，安全状态感知客户端需要周期调用查询接口函数SPI_GetProperty(“SysQuery:Task-Result”,…)，并轮巡到该函数成功返回0就说明本次任务执行完成，并通过SPI_GetProperty的第二个参数带回任务的执行结果。执行结果为升级后病毒库定义版本信息，格式如下：

<queryOutput operation=”AVTask:Virus-Def-And-Engine-Update”>

<simpleResult>result</simpleResult>

</queryOutput>

此处operation的值就是本次任务对应的任务名称，而result处填置为病毒库升级后的定义日期以及杀毒引擎升级后的版本，具体格式为：“Virus-Def-Version:yyyy-mm-dd;Virus-Engine-Version:yyyy-mm-dd”。queryOutput标签下只能包含一个simpleResult标签。

A.5.4 AVTask:System-Patch-Update

作用

提交升级操作系统补丁任务。

1）参数

无。

2）返回值

0——操作成功；

101——指定的任务不支持/不存在。

当返回值为0时，安全状态感知客户端需要周期调用查询接口函数SPI_GetProperty(“SysQuery:Task-Result”,…)，并轮巡到该函数成功返回0就说明本次任务执行完成，并通过SPI_GetProperty的第二个参数带回任务的执行结果。执行结果反映本次升级打上的补丁以及还需要打上的补丁。格式如下：

<queryOutput operation=”AVTask:System-Patch-Update”>
<complexResult>
<patch>
<item type=”add”>
<value>patch1</value>
<value>patch2</value>
</item>
<item type=”need”>
<value>patch3</value>
<value>patch4</value>
</item>
</patch>
</complexResult>
</queryOutput>

queryOutput标签的operation属性的值就是本次任务对应的任务名称，queryOutput标签下包含一个complexResult标签，complexResult标签下又包含一个patch标签，patch标签下包含两个item标签（type分别为add和need），一个item标签下可以包含0个、1个或多个value标签。一个value标签的取值为操作系统补丁的名称，type分别为add和need的两个item标签分别表示本次打上的补丁和还需要打上的补丁。

注意：本功能由AV插件供应商根据自己情况选择实现、部分实现或不实现。所谓部分实现是指不实现升级OS补丁任务，但是根据当前配置的安全策略检查并反馈系统缺少哪些补丁，即执行结果的格式为：

<queryOutput operation="AVTask:System-Patch-Update">
<complexResult>
<patch>
<item type="need">
<value>patch3</value>
<value>patch4</value>
</item>
</patch>
</complexResult>
</queryOutput>

A.6 插件配置文件描述

插件配置文件的格式如下（配置项的名称及值区分大小写）。

[MainInfo]

插件动态链接库名称。用于验证插件配置文件与DLL文件相匹配，该值与对应DLL文件名称应该一致。

PluginDllName=av-plugin.dll

插件名称 。用于验证插件配置文件与DLL文件相匹配，该值与对应DLL文件中的“产品名称”属性的值应该一致。

ProductName=CCIA.av-plugin

插件动态链接库版本。用于验证插件配置文件与DLL文件相匹配，该值与对应DLL文件中的“产品版本”属性的值应该一致。

PluginVersion=1.0.0.0

插件类型 。目前只有Plugin_AV。

PluginType=Plugin_AV

[QueryInfo]

插件提供的状态查询功能列表（格式为PostureQuery.功能号）

6.4描述的通用状态查询功能，无需在此指定；6.5描述的状态查询功能，如果支持，需要在此指定。

PostureQuery.1=AVQuery:Virus-Def-Version

PostureQuery.2=AVQuery:Engine-Version

……

[TaskInfo]

插件提供的执行任务功能列表（格式为TaskFunc.功能号）

6.4描述的通用执行任务功能，无需在此指定；6.5描述的执行任务功能，如果支持，需要在此指定。

TaskFunc.1=AVTask:Kill

TaskFunc.2=AVTask:Strategies-Update

……

[ErrorCodeInfo]

插件的错误号及多语言错误描述信息定义

ErrorCode.10001.zh_CN= AV客户端正在杀毒

ErrorCode.10001.en_US= AV client is busy killing virus

ErrorCode.10002.zh_CN= AV客户端正在升级

ErrorCode.10002.en_US= Updating AV client

……

中华人民共和国

通信行业标准

IP网络端点准入控制框架技术要求

YD/T 2172-2010

*

人民邮电出版社出版发行

北京市崇文区夕照寺街14号A座

邮政编码：100061

北京新瑞铭印刷有限公司印刷

*

开本：880×1230 1/16　　2011年2月第1版

印张：2　　2011年2月北京第1次印刷

字数：52千字

ISBN 978-7-115-2191/11-142

定价：20元