SIEMENS西门子 3VA1 IEC断路器 3VA1 112-5ED32-0AA0

产品名称	SIEMENS西门子 3VA1 IEC断路器 3VA1 112-5ED32-0AA0
公司名称	浔之漫智控技术(上海)有限公司
价格	.00/件
规格参数	西门子:代理经销商 低压断路器:全新原装 假一罚十 德国:正品现货 实体经营
公司地址	上海市松江区石湖荡镇塔汇路755弄29号1幢一层A区213室
联系电话	15801815554 15801815554

产品详情

通过加密确保数据机密
消息加密是数据安全的一项重要措施。在通信过程中，即使加密的消息被第三方截获，这些潜在的侦听者也无法访问所获取的信息。 在进行消息加密时，采用了大量的数学处理机制（算法）。所有算法都通过一个“ 密钥 ”参数，对消息进行加密和解密。 算法 + 密钥 + 消息 => 密文 密文 + 密钥 + 算法 =>（明文）消息 对称加密
对称加密的关键在于，两个通信伙伴都采用相同的密钥对消息进行加密和解密，如下图所 示：Bob 使用的加密密钥与 Alice 使用的解密密钥相同。即，我们常说的双方共享一个密钥，可通过该密钥对消息进行加密和解密。① Bob 采用对称密钥对消息进行加密 ② Alice 采用对称密钥对加密后的消息进行解密对称加密
该过程类似于一个公文箱，发送方和接收方使用同一把钥匙打开或锁上该公文箱。
优势：对称加密算法（如，AES、Advanced Encryption Algorithm）的速度较快。
缺点：如何将密钥发送给接收方，而不会落到其他人手中？此为密钥分发问题。如果截获的消息数量足够大，则可推算出所用的密钥，因此必须定期更换。
如果通信伙伴比较多，则需分发的密钥数量巨大。 非对称加密
在非对称加密技术中使用一对密钥：一个公钥和一个私钥。与 PKI 一同使用时，又称为公钥加密系统，简称 PKI 加密系统。通信伙伴（下图中的 Alice）拥有一个私钥和一个公钥。公钥对所有人公开。即，任何通信伙伴都可以获得该公钥。拥有公钥的通信伙伴可对发送给 Alice 的消息进行加密。即下图中的 Bob。① Alice 将其公钥提供给 Bob。无需采取防范措施即可实现该过程：只要确定采用的是 Alice 的公 钥，所有人都可以发消息给 Alice。 ② Bob 使用 Alice 的公钥对消息进行加密。 ③ Alice 使用私钥对 Bob 发送的密文进行解密。由于仅 Alice 拥有私有且未公开，因此只有她才能 对该消息进行解密。通过私钥，Alice 可以对使用她所提供的公钥加密的消息进行解密，而不仅 仅只是 Bob 的消息。 图 4-8 非对称加密
该系统与邮箱类似，所有人都可以向邮箱发送消息，但只有拥有密钥的人才能删除这些消息。

优势：使用公钥加密的消息，仅私钥拥有者才能进行解密。由于在解密时需要使用另一密钥（私钥），而且加密的消息数量庞大，因此很难推算出解密密钥。这意味着，公钥无需保持机密性，而这与对称密钥不同。
另一大优点在于，公钥的发布更为方便快捷。在非对称密钥系统中，接收方将公钥发送到发送方（消息加密方）时无需建立专用的安全通道。与对称加密过程相比，密钥管理工作 量相对较少。
缺点：算法复杂（如，RSA，以三位数学家 Rivest、Shamir 和 Adleman 的名字的首字母命名），因此性能低于对称加密机制。 实际通信中的加密过程 在实际通信过程中（如，与 CPU Web 服务器通信和开发式用户安全通信），通常在相关的应 用层之后使用 TLS 协议。例如，应用层采用的协议为 HTTP 或 SMTP，详细信息见前文所述。 例如，TLS (Transport Layer Security) 混合采用非对称加密和对称加密（混合加密）机制确保 数据通过 Internet 进行安全传输，并支持以下子协议： TLS Handshake Protocol，对通信伙伴进行身份验证，并在非对称加密的基础上对数据传输 所需的算法和密钥进行协商 TLS Record Protocol 采用对称加密机制对用户数据加密以及进行数据交换。
无论是非对称加密还是对称加密，这两种数据安全加密机制在安全性方面没有明显差异。数据安全等级取决于设置的参数，如所选密钥的长度等等。 加密使用不当 通过位串，无法指定公钥的身份。欺瞒者可使用他们自己的公钥声明为其他人。如果第三方使用该公钥将其认作是指定的通信伙伴，则将导致机密信息被窃取。之后，欺瞒者再使用自己的密钥对这些本消息进行解密，虽然这些消息本不应发送给他们。最终，导致敏感信息泄露，落入他人之手。

为了有效预防此类错误的发生，该通信伙伴必须确信与正确的通信伙伴进行数据通信。此类信任关系是通过 PKI 中的数字证书建立的。通过签名确保数据的真实性和完整性 由能够截获服务器与客户端之间的通信并将自身伪装成客户端或服务器的程序实施的攻击称为中间人攻击。如果未能检测到这些程序的真实身份，则将造成诸如 S7 程序、CPU 中设定值等重要信息泄漏，进而导致设备或工厂遭受攻击。可使用数字证书避免此类攻击。
在安全通信过程中，所用的数字证书符合 International Telecommunication Union (ITU) 的 X.509 标准。该证书用于检查（认证）程序、计算机或组织机构的身份。 如何通过证书建立信任关系 X.509 证书主要用于将带有证书的数据身份（如，电子邮件地址或计算机名称）与公钥中的身份绑定在一起。身份可以是个人、计算机，也可以是机器设备。 证书由证书颁发机构（Certificate Authority，CA）或证书主体签发。而 PKI 系统则指定了用户 信任证书颁发机构及其所签发证书的规则。
证书认证过程： 1. 要获取一份证书，需要向与证书颁发机构相关联的注册机构提交一份证书申请。 2. 证书颁发机构将基于既定标准对该申请和申请人进行评估。 3. 如果可以清晰识别申请人的身份，则证书颁发机构将签发一份已签名的证书进行确认。申请人现成为证书主体。 在下图中，对这一过程进行了简要说明。但不涉及 Alice 对该数字签名的检查过程。自签名证书
自签名证书指，由证书主体而非独立的证书颁发机构签名的证书。 示例：
用户也可以自己创建证书并签名，对发送给通信伙伴的消息进行加密。在上述示例中， Bob（而非 Twent）可以使用私钥对自己的证书进行签名。之后，Alice 将使用 Bob 的公钥 检查该签名是否与 Bob 的公钥相匹配。该过程可用于简单的工厂内部数据加密通信。 例如，根证书是一种由证书颁发机构 (CA) 签署的自签名证书，其中包含证书颁发机构的公 钥。自签名证书的特性
自签名证书的证书主体“ CN ”(Common Name of Subject) 和“ Issuer ”属性相同：用户已完成对证书的签名。字段“ CA ”(Certificate Autority) 需设置为“ False ”；自签名证书不得用于对其它证书进行签名。 自签名证书未包含在 PKI 系统中。 证书内容 符合 X.509 V3 标准（同样用于 STEP 7 和 S7-1500 CPU）要求的证书通常包含以下元素： 公钥 证书主体（即，密钥持有者）的详细信息。如，Common Name (CN) of Subject 。 各种属性，如序列号和有效期等等 证书颁发机构 (CA) 的数字签名，用于证实信息的正确性。 除此之外，还包含以下扩展详细： 指定公钥的使用范围(Key Usage)，如签名或密钥加密。 在开放式用户安全通信中，使用 STEP 7 创建一个新证书时，可从用途列表中选择相应的条 目，如“ TLS ”。 指定 Subject Alternative Name (SAN)，用于与 Web 服务器进行安全通信 (HTTP over TLS)，以确保 Web 浏览器地址栏中的证书同样属于该 URL 所指定的 Web 服务器。 如何生成并验证签名

非对称密钥可用于证书的验证：在“ MyCert ”证书示例中，介绍了具体的“ 签名 ”与“ 验证签名 ”过程。 生成签名： 1.“ MyCert ”证书的签发者使用一个特定的哈希函数（例如，SHA-1，Secure Hash Algorithm），根据证书数据生成一个哈希值。 该 HASH 值是一个长度固定的位串。HASH 值长度固定的优势在于，签名的时间始终相 同。 2. 之后，证书的签发者再使用由这种方式生成的 HASH 值和私钥，生成一个数字签名。通常 采用 RSA 签名机制。 3. 数字签名将保存在证书中。此时，证书已签名。 验证一个签名： 1. “ MyCert ”证书的认证方将获得签发者签发的证书和公钥。 2. 使用签名时所用的哈希算法（例如，SHA-1），根据证书数据生成一个新的哈希值。 3. 最后，再将由证书签发者公钥确定的 HASH 值与签名算法进行比较，对签名进行检查。 4. 如果签名通过检查，则表示证书主体的身份以及完整性（即，证书内容的可靠性和真实性）均通过验证。拥有该公钥（即，证书颁发机构的证书）的任何人均可对该签名进行检查，并确认该证书确实由该证书颁发机构签发。签名消息 上文中介绍的证书签名与验证机制，同样使用 TLS 会话对消息进行签名和验证： 如果发送方基于一条消息生成一个 HASH 值并使用私钥进行加密，之后在添加到原消息中，则 消息接收方即可对消息的完整性进行检查。接收方使用发送方的公钥对该 HASH 值进行解密，并将其与所收到消息中的 HASH 进行比较。如果这两个值不同，则表示该消息或加密的 HASH 值在传送过程中被篡改。 Root 证书的证书链 PKI 证书通常按层级进行组织：层级顶部由根证书构成。Root 证书并非由上一级证书颁发机构 签名。Root 证书的证书主体与证书的签发者相同。根证书享受juedui信任。它们构成了信任“ 点 ”，因此可作为接收方的可信证书。此类证书存储在专门存储受信证书的区域。 基于该 PKI，Root 证书可用于对下级证书颁发机构颁发的证书（即，所谓的中间证书）进行签 名。从而实现从 Root 根证书到中间证书信任关系的传递。由于中间证书可对诸如 Root 证书之类的证书进行签名，因此这两种证书均称为“ CA 证书 ” 这种证书签名层级可通过多个中间证书进行延伸，直至最底层的实体证书。最终实体证书即为待识别用户的证书。 验证过程则反向贯穿整个层级结构：综上所述，先通过签发者的公钥确定证书签发者并对其签名进行检查，之后再沿着整条信任链确定上一级证书签发者的证书，直至到达根证书。 结论：无论组态何种安全通信类型，每台设备中都必需包含一条到 Root 证书的中间证书链（即证书路径），对通信伙伴的最低层实体证书进行验证。