Title: SYSTEM AND METHOD FOR AUTHENTICATING COMPONENTS OF A STORAGE NETWORK

Abstract: A network includes one or more hosts that make access requests to read data from or write data to one or more storage devices. The hosts are coupled to the storage devices through a fabric that includes a plurality of switches. At least some of the components of the network (the hosts, switches and storage devices) are configured to authenticate other components before communicating data to or accepting data from the other components. The authenticating components, unauthenticating components are prevented from accessing data written to, read from or stored on the storage devices. In one embodiment, public key security is used to initially authenticate pairs of components, which then subsequently communicate using a shared secret symmetric key.
Title: System and Method for Authenticating Components of
A Storage Network

Field of the Invention

[0001] This invention relates to devices in a storage network. More particularly, the invention relates to a system and method for authenticating such devices.

Background of the Invention

[0002] Many modern computer systems include numerous storage devices, each of which may be accessed by one or more host computers. For example, enterprise computer systems will typically include a number of storage devices such as data storage arrays consisting of multiple hard disk drive, optical driver (CD and DVD based drives which can be based on read-only, write once or rewritable standards), solid state storage devices such as flash memory, and other types of magnetic and optical storage devices.

[0003] Such enterprise systems may also include various servers, desktop user computers and other devices which may require access to portions of the data stored on the storage devices. Typically, these devices, which may be collectively referred to as hosts, are indifferent as to the physical location of the data space that they are attempting access for either read or write operations. The host devices will typically refer to the data space by a logical name such as "John Doe's server" or "Marketing Group Folder". A storage area network (SAN) utilizes a "fabric" to facilitate transparent access for a host to different data spaces, regardless of the physical location of the data spaces. These physical locations may change as the storage needs of the enterprise change. The fabric typically includes one or more switches that translate the logical name used by the host for a data space into the physical location of the data space, and
provide a connection between the host and the data space. Some or all of the switches are aware of the topology of the network, including the physical interconnections between switches, between hosts and switches and between switches and storage devices. The switches pass access requests, data and associated information such as read and write confirmation messages between the host and the data space.

[0004] As such information passes between switches and through switches, it may be vulnerable to unauthorized duplication or theft. For example, an unauthorized person may introduce a rogue switch into the fabric. The rogue switch may be configured to operate like other switches, but it may additionally be configured to transmit a copy of all or some of the information passing through it to another computer, which is under the control of the unauthorized person.

Similarly, an unauthenticated or unauthorized user may use a rogue host or a rogue storage device to copy or steal data.

[0005] It is preferable that the switches in the network communicate access requests, data and other information only to other switches that are authenticated components of the network. It is also desirable that the switches provide data access services only to authenticated hosts. It is also desirable that the switches provide access only to authenticated storage devices.

[0006] Accordingly, there is a need for a system and method for authenticating components of a storage network, such as a storage area network or a network attached storage..

Summary of the Invention

[0007] The present invention provides a mechanism for allowing two components in a storage network to authenticate each other to enable secure communication of data and other information between them. The first component selects a random number, generates a public value based on the random number and a digital signature based on the public value and a private key held by the first
component. The first component transmits the public value, digital signature and public key corresponding to its private key. The public key may be embodied in a public key certificate issued by a certificate authority that may be part of the network.

[0008] The second component authenticates the first component as an authentic component of the network using the information it received from the first component. If the first component's public key is embodied in a public key certificate, the second component may check the authenticity of the certificate with the certificate authority.

[0009] The second component also follows a similar procedure of generating a random number, a public number and a digital signature and the first component authenticates the second component as an authentic component of the network in a similar manner.

[0010] Each component that successfully authenticates the other component generates a session key for use in communicating with the other component. If both components successfully authenticate each other, they generate the same session key, which is then a shared secret symmetric key and which may then be used to secure communications between the components.

[0011] Each component may also have an access control list associated with it. Before entering into communication with the other component, each component also checks its own access control list to ensure that the other component is permitted to access it. If each component is permitted to access the other component, the components establish a secure communication session using the shared secret symmetric key.

[0012] In other embodiments of the invention, different authentication techniques may be used. In addition, different methods of sharing a symmetric key may be used.
In one aspect, the present invention provides, in a
network used for data storage, a method for a first component and a
second component to establish a shared secret session key for use
in subsequent data transfers, wherein each of the first and second
component has a public key and a complementary private key, the
method comprising:

(a) the first component selecting a first random value \( x \);
(b) the first component calculating a first public value \( w_1 \)
using the first random value;
(c) the first component calculating a first digital signature \( s_1 \)
using the first public value and the first component's
private key;
(d) the first component transmitting first public value, the
first digital signature and the first component's public
key to the second component;
(e) the second component selecting a second random
value \( y \);
(f) the second component calculating a second public
value \( w_2 \) using the second random value;
(g) the second component calculating a second digital
signature \( s_2 \) using the second public value and the
second component's private key;
(h) the second component transmitting the second public
value, the second digital signature and the second
components public key to the first component;
(i) the first component attempting to authenticate the
second component using the second public value and
the second component's public key;
(j) the second component attempting to authenticate the
first component using the first public value and first
component's public key.
[0014] The first and second component may be any type of components in the network, including hosts, switches and storage devices. They first and second components may be different by types of components.

5 Brief Description of the Drawings

[0015] A preferred embodiment of the present invention will now be described in detail with reference to the drawings, in which like reference numerals indicate like parts throughout the several Figures. In the drawings:

10 Figure 1 is schematic description of a network incorporating components configured according to the present invention;

Figure 2 is a more detailed schematic description of a fabric of Figure 1;

15 Figure 3 is a flowchart illustrating a method of authenticating components of the network of Figure 1.

Detailed Description of Exemplary Embodiments of the Invention

[0016] Reference is first made to Figure 1, which illustrates an exemplary network 100 that incorporates an exemplary fabric 102 according to the present invention.

[0017] Network 100 includes one or more host devices 104 which are coupled to the fabric 102 through connectors 106. Network 100 also includes one or more storage devices 108 that are coupled to fabric 102 through a set of connectors 109.

[0018] Network 100 may be a storage area network (SAN) or a network attached storage or any other type of network that includes hosts, switches and storage devices.

[0019] Hosts 104 may include server computers, or other devices that require access to data stored on storage devices 108. The connector 106 used to couple each host 104 to fabric 102 is selected to be compatible with both host 104 and fabric 102. For
example, a connector may be a host bus adapter for use with a Fibre Channel network or a network interface card for use with an internet protocol based network such as the Internet.

[0020] Fabric 102 allows each host 104 to form a connection with one or more of storage devices 108. The connection is created and, in one embodiment, persists through the fabric for the time required to satisfy (or deny) a request by a host 104 to access a storage device 108.

[0021] Storage devices 108 may be magnetic, optical or other types of storage devices, including magnetic hard or floppy disk drives and CD and DVD based storage devices that may be used to store computer readable information, including computer programs and data. All such computer readable information is referred to here generically as data.

[0022] Reference is next made to Figure 2, which illustrates fabric 102 in greater detail. In this exemplary embodiment, fabric 102 comprises a number of switches 110. Switches 110a – 110c are coupled to hosts 104a – 104e. Typically, each host will be able to directly communicate with one or more switches 110. Switches 110x – 110z are coupled to storage devices 108a – 108d. Typically, each storage device 108 will be coupled directly to one or more switches 110.

[0023] System 100 may include additional hosts 104 and storage device 108, each of which may be coupled to any of switches 110. A particular switch 110 may be simultaneously coupled to one or more hosts 104 and one or more storage devices 108. A single computing device may be used to operate multiple hosts 104, multiple switches 110 and may include multiple storage devices 108.

[0024] In operation, hosts 104 make requests for access to storage devices 108. Such access requests may be read data from or write data to the storage devices 108. Switches 110 are configured to produce a temporary, but persistent connection between a
requesting host, and the storage device for which the host has made an access request. For example, if host 104a makes a read data request to access data stored on storage device 108d, then switches 110a, 110e, 110i and 110z form a persistent connection between host 104a and storage device 108d. The connection is maintained as long as it is required to permit the read data request to be satisfied (or denied, if it is not properly made). When the request has been satisfied, the connection through switches 110a, 110e, 110i and 110z is terminated. Typically, each switch 110 may simultaneously be part of more than one persistent connection between a host 104 and a storage device 108. In addition, each host 104 may simultaneously be connected to more than one storage device 110. Each storage device 110 may be connected and storage device 110 may be connected to more than one host 104. For example, the following connections may exist simultaneously:

i. a connection between host 104a and storage device through switches 110a, 110e, 110i and 110z,

ii. a connection between host 104c and storage device through switch 108a through switches 110c, 110g, 110i and 110y; and

iii. a connection between host 104a and storage device 108a through switches 110a, 110d, 110h and 110x.

Each of these connections is independently created and then terminated as required to permit various requests from hosts 104 for access various storage devices 108 to be satisfied.

During each request by a host 104 to access a storage device 108, data is transmitted through the switches that form the connection between the host and the storage device. It is desirable to ensure that such data is not intercepted by unauthenticated person.

One method an unauthenticated user may use to intercept data is to insert a rogue switch 110r into fabric 102, as
shown in Figure 2. Rouge switch 110r may participate in connection between hosts 104c and 104d to storage devices 108c and 108d. Rogue switch 110r may be configured to operate in the same manner as the other switches 110 to permit connections between hosts 104 and storage device 108. However, rogue switch may also record some or all of the data passing through it, or transmit a copy of some or all of the data passing through it a computer R, operated by the unauthenticated user. The unauthenticated user may then access the data by reviewing the data stored or transmitted by the rogue switch 110r.

[0027] It is desirable to prevent such rogue switches from participating in connections between hosts 104 and storage devices 108. This embodiment of the present invention utilizes a key-based security protocol to authenticate switches (and other components) in fabric 102. Each switch 110 in fabric 102 is configured to authenticate every other switch 110 that it communicates with prior to forming a connection with that switch.

[0028] Reference is next made to Figure 3, which illustrates an authentication method 200 by which a pair of switches 110, or any other pair of components in network 100, may authenticate one another in order to establish secure communications between the two switches. Method 200 has two execution threads, one of which is carried out at each of the pair of components. Each switch in a connection performs method 200 in respect of every switch that is adjacent to it in a connection between a host 104 and a storage device 108, prior to participating in the connection. Method 200 is described here in the context of switch 110a authenticating switch 110e prior to establishing communications with switch 110e.

[0029] Each switch 104 in fabric 102 has a public key and a complementary private key. Each switch’s public key is set out in a public key certificate 112 issued to the switch by a certificate authority 114. The dashed line between certificate authority 114 and switch
110a indicates that certificate authority 114 is able to communicate with switch 110a. Certificate authority 114 is similarly able to communicate with the other devices in network 100. Each switch maintains its private key confidential from the other components of network 100. Every switch in fabric 102 also has a public key of the certificate authority. The certificate has a corresponding private key that it keeps confidential. In Figure 2, only the public key certificate 112a of switch 110a is illustrated. Each of the other switches 110 has a corresponding public key certificate.

[0030] A particular switch's private key may be used to decrypt information that has been encrypted using the switch's public key. The switch may thus obtain encrypted information from another component of network 100 (or any other device that is capable of communicating with the switch) and decrypt it.

[0031] Similarly, a particular switch's public key may be used to decrypt information that has been encrypted using the switch's private key. The switch may thus transmit encrypted information that other components may decrypt using the switch's public key. This may be used by the switch to "sign" a data transmission. Briefly, the switch may transmit an unencrypted signature (which may be a string of characters or other data), as well as a encrypted version of the signature that has been encrypted using the switch's private key. Another device may decrypt the encrypted version of the signature information. If the decrypted signature matches the unencrypted signature, the other device knows that the encrypted signature was in fact encrypted using the switch's private key and, based on the assumption that each switch's private key is kept confidential by the switch, may assume that the message originated from the switch.

[0032] In addition, each switch 110 in fabric 102 maintains securely or can securely access an access control list (ACL) 118 which identifies all other components of network 100 that are permitted to access (or communicate with) the switch 110. In Figure
2, only access control 118a is shown in switch 110a. The other components of network 100 also have access control lists.

[0033] Method 200 begins in step 202, in which switch 110a transmits an "Initiate Authentication" message to switch 110e.

Method 200 then proceeds in two branches. Switch 110a proceeds at step 204 and switch 110e proceeds at step 222.

[0034] In step 204 switch 110a generates a random number x and generates a first public value $w_a = g^x$. The term g is a generator of a finite cyclic group of large prime order. The group, the order and the value of g is available to all components in network 100. This may be accomplished in various ways. For example, this information may be resident in the memory of all of the components, it may be provided with each public key certificate 112 or it may be available from the certificate authority 114. Switch 110a then proceeds to step 206.

[0035] In step 206, switch 110a uses its private key to generate an encrypted version $s_a$ of the first public value $w_a$. The encrypted version $s_a$ is used by switch 110a as its signature during method 200. Switch 110a then proceeds to step 208.

[0036] In step 208, switch 110a transmits the first public value $w_a$, the encrypted version $s_a$ of the first public value, and its public key certificate 112a to switch 110e.

[0037]

[0038] Switch 110e begins method 200 in step 224. In steps 224, 226 and 228, switch 110e follows a process similar to that followed by switch 110a in steps 204, 206 and 208.

[0039] In step 224, switch 110e generates a random number y and generates a second public value $w_e = g^y$. In step 226, switch 110e uses its private key to generate an encrypted version $s_e$ of the second public value $w_e$. The encrypted version $s_e$ is used by switch 110e as its signature during method 200. In step 228, switch 110e transmits the second public value $w_e$, the encrypted version $s_e$ of the second public value and its public key certificate 112e to switch 110a.
After step 208, switch 110a proceeds to step 210, in which it receives the second public value $w_e$, the encrypted version $s_e$ of the second public value and the public key certificate 112e of switch 110e.

Switch 110a attempts to authenticate switch 110e as an authenticated component of fabric 102 as follows. First, switch 110a authenticates public key certificate 112e by confirming that it was properly issued by certificate authority 114 to switch 110e. Switch 110a communicates securely with certificate authority 114 by using the public key of certificate authority 114.

If public key certificate 112e is not authentic, then switch 110a proceeds to step 236.

If public key certificate 112e is authentic, then switch 110a uses the public key set out in public key certificate 112e to decrypt the encrypted version $s_e$ of the second public value $w_e$, thereby producing a decrypted version $w_{e-d}$ of the second public value $w_e$. If the decrypted version $w_{e-d}$ is identical to the version of the second public value $w_e$ received from switch 110e, then switch 110a knows that switch 110e used its private key to encrypt the second public value $w_e$ and accordingly that switch 110e is an authenticated component of fabric 102.

If the decrypted version $w_{e-d}$ is not identical to the version of the second public value $w_e$ received from switch 110e, then switch 110e is deemed not to be an authenticated component of fabric 102.

If switch 110e is an authenticated component of fabric 102, then switch 110a proceeds to step 212. If switch 110e is not an authenticated component of fabric 102, then switch 110a proceeds to step 216.

In step 212, switch 110a calculates a session key $K_1$ for use in subsequent communications with switch 110e. Typically, switch 110a will use its first random number $x$ and the second public value $w_e$ to generate session key $K_1$. The algorithm used to generate
the session key $K_1$ is also known to switch 110e (which may use it in step 232 below). The algorithm permits switch 110e to generate an identical session key $K_1$ using the information available to it. Switch 100a next proceeds to step 214.

5. In step 214, switch 100a checks its access control list 118a to determine if switch 110e is permitted to access switch 110a. If switch 110e is listed in access control list 118a, then switch 110a has completed method 200. Otherwise, switch 110a proceeds to step 218.

10. In step 216, switch 110a identifies switch 110e as an unauthenticated device and will not transmit data to or accept data from switch 110e. Subsequently, switch 110a may repeat method 300 (at its own request, at the request of switch 110e, or at the request of an external user or other device) in respect of switch 110e. If switch 110e is authenticated at that time, then switch 110a may initiate communications with it.

15. In step 218, switch 110a identifies switch 110e as an unauthorized component and and will not transmit data to or accept data from switch 110e. As in the case when switch 110e is unauthorized, switch 110e may be authorized and authenticated by switch 110a at a later time by executing method 200 again.

20. After completing step 214 or 216 or 218, switch 110a has completed method 200.

25. Switch 110e also attempts to authenticate switch 110a in an analogous process. After step 228, switch 110e proceeds to step 230, in which it receives the first public value $w_a$, the encrypted version $s_a$ of the first public value and the public key certificate 112a of switch 110a.

30. Switch 110e attempts to authenticate switch 110a as an authenticated component of fabric 102 as follows. First, switch 110e authenticates public key certificate 112a by confirming that it was properly issued by certificate authority 114 to switch 110a. Switch
110e communicates securely with certificate authority 114 by using the public key of certificate authority 114.

[0053] If public key certificate 112a is not authentic, then switch 110e proceeds to step 236.

[0054] If public key certificate 112a is authentic, then switch 110e uses the public key set out in public key certificate 112a to decrypt the encrypted version $s_a$ of the first public value $w_a$, thereby producing a decrypted version $w_{a-d}$ of the first public value $w_a$. If the decrypted version $w_{a-d}$ is identical to the version of the first public value $w_a$ received from switch 110a, then switch 110e knows that switch 110a used its private key to encrypt the first public value $w_a$ and accordingly that switch 110a is an authenticated component of fabric 102.

[0055] If the decrypted version $w_{a-d}$ is not identical to the version of the first public value $w_a$ received from switch 110a, then switch 110a is deemed not to be an authenticated component of fabric 102.

[0056] If switch 110a is an authenticated component of fabric 102, then switch 110e proceeds to step 232. If switch 110a is not an authenticated component of fabric 102, then switch 110e proceeds to step 236.

[0057] In step 232, switch 110e generates the same session key $K_1$ that was calculated by switch 110 in step 212. Both switches 110a and 110e now have session key $K_1$, which they both keep secret. Switch 110e next proceeds to step 234.

[0058] In step 234, switch 100e checks its access control list 118e to determine if switch 110a is permitted to access switch 110e. If switch 110a is listed in access control list 118e, then switch 110e has completed method 200. Otherwise, switch 110e proceeds to step 238.

[0059] In step 236, switch 110a identifies switch 110a as an unauthenticated device and will not transmit data to or accept data from switch 110a. Subsequently, switch 110e may repeat method
300 (at its own request, at the request of switch 110a, or at the request of an external user or other device) in respect of switch 110a. If switch 110a is authenticated at that time, then switch 110e may initiate communications with it.

5

[0060] In step 238, switch 110e identifies switch 110a as an unauthorized component and will not transmit data to or accept data from switch 110a. As in the case when switch 110a is unauthorized, switch 110a may be authorized and authenticated by switch 110e at a later time by executing method 200 again.

10

[0061] After completing step 232 or 236 or 238, switch 110e has completed method 200.

[0062] In another embodiment of the present invention, switch 110a may simply begin method 200 at step 204 and carry out steps 204, 206 and 208. Switch 110e may treat receipt of the first public value $w_a$, the encrypted version $s_a$ of the first public value, and its public key certificate 112a as the "Initiate Authentication" message, and then carry out steps 224, 226 and 228 and then proceed directly to step 230, described below.

[0063] At the conclusion of method 200, one of several situations may exist. If each switch 110a and 110e authenticates the other switch as an authenticated components of network, then each switch will have an identical session key $K_1$. The switches may then communicate with each other using key $K_1$ as a symmetric key. Each switch encrypts data using key $K_1$ and transmits it to the other switch. The other switch is able to decrypt the data using key $K_1$. The switches may continue the use of key $K_1$ for any period of time. Typically, key $K_1$ will be used for a limited period of time. If the communication session between switches 110a and 110e ends, then both switches will typically be configured to require a new authentication using method 200 to generate a new session key $K_n$. This new key will be used for subsequent communication between the switches. Either or both of the switches may optionally be
configured to require a new authentication after a selected amount of time. For example, switch 110a may be configured to discard session key K, one hour after obtaining it. After that time, before switch 110a will transmit data to switch 110e, it will initiate a new authentication using method 200.

[0064] If either or both of switch 110a or switch 110e is unable to authenticate the other switch, then the two switches will be unable to communicate with one another using a symmetric key (since one or both will not have generated the symmetric key). In this case, the two switches do not transmit data to one another until they are able to authenticate each other by subsequent carrying out method 200.

[0065] Method 200 may be used by any pair of switches 110 in fabric 102 to authenticate one another. Each switch 110 maintains a list of other switches that it has authenticated along with the session key K to be used for communication with each authenticated switch. Each session key K will normally be different and each pair of switches that are sharing a session key K keep it secret from all other switches.

[0066] Reference is again made to Figure 1. Typically each switch 110 in fabric 102 will initiate method 200 when it needs to initiate communications with a switch that it does not presently have a session key K for. As hosts 104 make requests to access storage devices 108, each switch through which the required connection will be made uses an existing session key K or generates a new session key K to communicate with its adjacent switches. Once the connection has been established, the host will transmit its request through the switches participating in the connection to the data storage device. Between each pair of adjacent switches in the connection the request is encrypted and decrypted using the appropriate session key K. Similarly data and other information (such and read and write confirmations, etc.) are encrypted and
decrypted as they pass between each pair of adjacent switches in the connection.

[0067] Thus far, the present invention has been described to illustrate its use in authenticating switches 110 that form fabric 102. The present invention may also be used to authenticate devices other components of network 100 including hosts 104 and storage devices 108. Each such device may obtain a public key certificate from certificate authority 114 and may use a complementary pair of public and private keys to be authenticated by switches 110 and to authenticate switches 110. In such an embodiment of the present invention, each transmission of a request by a host 104 to access a storage device 108 will occur only between pairs of devices that have authenticated one another.

[0068] Referring to the previous example in which host 104a made a read data request for data stored on storage device 108d and a persistent connection is established through switches 110a, 110e, 110i and 110z, each of the switches will have authenticated the adjacent switches in the connection before entering into the connection to transfer the request and any data between host 104a and storage device 108d. Host 104a and switch 110a are using session key $K_2$ to communicate. Switches 110a and 110e are using session key $K_1$ to communicate. Switches 110e and 110i are using session key $K_3$ to communicate. Switches 110i and 110z are using session key $K_4$ to communicate. Switch 110z and host 108d are using session key $K_5$ to communicate.

[0069] In Figure 1, the communication links between switches 110, and between the hosts and switches, and between the switches and storage devices 108 are illustrated as direct connections. These connection may also be indirect and may be through other devices, which may or may not be switches. For example, two switches may connect with each other through the Internet or through an intranet with other devices and computers interposed between them.
[0070] Method 200 is described above using a public key security protocol to initially authenticate components of network 100. If both devices authenticate one another, they calculate shared secret session key K using the Diffie-Hellman algorithm. The session key K is subsequently used for symmetric key security.

[0071] Any other method of authentication may be used by components to authenticate one another within the scope of the present invention. Similarly, another method may be used for secure communications after authentication. For example, the components may continue to use public key encryption to communicate even after authentication, although, typically, symmetric key encryption requires less communication and computation overhead and as a result allows faster processing.

[0072] In another embodiment, after generating the session key K, the components using method 200 may conduct an explicit key confirmation by sending information derived from the session key K. For example, a random challenge which can be answered based on the session key K could be used to verify that both components have generated the same session key K.

[0073] Network 100 has been described as requiring adjacent components in a connection between a host 104 and a storage device to be individually authenticated to each other. In another network according to the present invention, components that are not adjacent to one another in a connection may authenticate each other, through some intermediate components. The mutually authenticated components may then communicate using a shared secret session key K.

[0074] System 100 has been described as having physical storage devices 108 connected to switches 110x to 110z. In other embodiments, the physical storage devices 108 may be logically divided or combined into logical units, which are then presented to the fabric 102 as individual storage volumes. One or more devices
that provides translation between the logical units and the physical structure of the storage units 108 would be interposed between switches 110x, 110y and 110z and storage devices 108.

[0075] In another embodiment of the present invention, switches 100 may not provide a persistent connection. Instead, they may operate with a packet switched protocol in which individual packets of information (which may include all or part of an access request or data that is to be written to or read from a storage device) are passed between switches and are routed to their intended destination. Such a packet switched protocol.

[0076] Switches 100 are typically bidirectional switches, and typically, they are capable of creating a connection in various arrangements. For example, host 104a may be connected to storage device through switches 110a, 110d, 110h, 110f, 110c, 110g, 110i and 110z. Similarly, if a packet switched protocol is used, packets may flow in different routes between their ultimate source and their ultimate destination.

[0077] In other embodiments, a connection between a host and a storage device may persist for longer than the time required to complete (or deny) a single access request. A connection may be maintained to allow a series of transactions to be performed, or may be maintained for a selected time, or until it has been unused for a selected time. A single embodiment may utilize one or more of these connection persistence policies.

[0078] System 100 includes a number of switches in fabric 102. In other embodiments, the network 100 may be much simpler or much more complex. For example, another network may include a single host 104, a single switch 110 and a single storage device 108. On the other hand, a network may include hundreds of hosts 104, switches 110 and storage devices 108.

[0079] In another embodiment of system 100, certificate authority 114 may maintain a list of certificates that have been
invalidated due to expiry (i.e. the certificate was valid only up to a past point in time) or for other reasons. When a component attempts to authenticate a certificate in step 210 or 230, then certificate authority may check its list of invalid certificates and refuse to authenticate such certificates. Certificate authority 114 may also provide additional service such as indicating problems with a certificate (i.e. name mismatch between the owner of the certificate and the device presenting it, past expiry date, prior to initial validity date and other problems). Such problems may be individually classified as fatal flaws that result in a failure to authenticate the certificate at all, or as less serious flaws, which are indicated to the component attempting to authenticate the certificate, along with an indication that the certificate was issued by the certificate authority.

[0080] The present invention has been described in an exemplary way. The invention is subject various additional modification and variations within its scope, which is limited only by the appended claims.
We claim:

1. In a network used for data storage, a method for a first component and a second component to establish a shared secret session key for use in subsequent data transfers, wherein each of the first and second component has a public key and a complementary private key, the method comprising:

   (a) the first component selecting a first random value \( x \);

   (b) the first component calculating a first public value \( w_1 \) using the first random value;

   (c) the first component calculating a first digital signature \( s_1 \) using the first public value and the first component's private key;

   (d) the first component transmitting first public value, the first digital signature and the first component's public key to the second component;

   (e) the second component selecting a second random value \( y \);

   (f) the second component calculating a second public value \( w_2 \) using the second random value;

   (g) the second component calculating a second digital signature \( s_2 \) using the second public value and the second component's private key;

   (h) the second component transmitting the second public value, the second digital signature and the second components public key to the first component;

   (i) the first component attempting to authenticate the second component using the second public value and the second component's public key;
(j) the second component attempting to authenticate the first component using the first public value and first component's public key;

2. The method of claim 1 further comprising, if the first component successfully authenticates the second component and if the second component successfully authenticates the first component, then each of the first and second components generating an identical session key.

3. The method of claim 1 or 2 wherein the first public value $w_1$ is equal to $g^x$ and wherein the second public value $w_2$ is equal to $g^y$, wherein $g$ is a generator of a finite cyclic group.

4. The method of any one of claim 1 to 3 wherein the first digital signature is calculated by encrypting the first public value $w_1$ using the first component's private key.

5. The method of claim 4 wherein the second digital signature is calculated by encrypting the second public value $w_2$ using the second component's private key.

6. The method of claim 1 wherein the first component is a host.

7. The method of claim 6 wherein the second component is a switch.

8. The method of claim 6 wherein the second component is a storage device.

9. The method of claim 1 wherein the first component is a switch.
10. The method of claim 9 wherein the second component is a storage device.

11. The method of claim 1 wherein the network is a storage area network.

12. The method of claim 1 wherein the network is a network attached storage.

13. The method of claim 1 further including the first component checking an access control list associated with the first component to determine whether the second component is permitted to access the first component.

14. The method of claim 13 further including the second component checking an access control list associated with the second component to determine whether the first component is permitted to access the second component.

15. The method of claim 1 further including, in response to each of the first and second components authenticating the other of the first and second components, each of the first and second components generating a session key for use in subsequent secure communication between the first and second components.
### INTERNATIONAL SEARCH REPORT

#### A. CLASSIFICATION OF SUBJECT MATTER
- IPC 7: H04L9/32, H04L9/08, H04L29/06, H04L9/30

According to International Patent Classification (IPC) or to both national classification and IPC.

#### B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
- IPC 7: H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
- EPO-Internal, WPI Data

#### C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>US 5 761 305 A (QU MINGHUA ET AL) 2 June 1998 (1998-06-02) abstract column 3, line 52 -column 4, line 37</td>
<td>1-15</td>
</tr>
<tr>
<td>X</td>
<td>EP 1 081 891 A (COMPAQ COMPUTER CORP) 7 March 2001 (2001-03-07) paragraph ‘0039! - paragraph ‘0055!</td>
<td>1-15</td>
</tr>
</tbody>
</table>

- Further documents are listed in the continuation of box C.
- Patent family members are listed in annex.

- Special categories of cited documents:
  - "A" document defining the general state of the art which is not considered to be of particular relevance
  - "E" earlier document but published on or after the international filing date
  - "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
  - "O" document referring to an oral disclosure, use, exhibition or other means
  - "P" document published prior to the international filing date but later than the priority date claimed
  - "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
  - "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
  - "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
  - "S" document member of the same patent family

Date of the actual completion of the international search: 23 July 2003
Date of mailing of the international search report: 05/08/2003

Name and mailing address of the ISA European Patent Office, P.B. 5816 Patentlaan 2 NL - 2280 HU Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016

Authorized officer: Apostolescu, R
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>US 5896455 A</td>
<td>20-04-1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO 9818234 A1</td>
<td>30-04-1998</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 5889865 A</td>
<td>30-03-1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AU 5266596 A</td>
<td>07-11-1996</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CA 2174261 A1</td>
<td>22-10-1996</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO 9633565 A1</td>
<td>24-10-1996</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP 0739105 A1</td>
<td>23-10-1996</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 6122736 A</td>
<td>19-09-2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 2001148697 A</td>
<td>29-05-2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AT 187588 T</td>
<td>15-12-1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CA 2130250 A1</td>
<td>18-02-1995</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 69327238 D1</td>
<td>13-01-2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 69327238 T2</td>
<td>07-09-2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 5600725 A</td>
<td>04-02-1997</td>
</tr>
<tr>
<td>US 6263445</td>
<td>17-07-2001</td>
<td>DE 19983352 T0</td>
<td>13-06-2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB 2354617 A ,B</td>
<td>28-03-2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 2002519781 T</td>
<td>02-07-2002</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO 0000881 A1</td>
<td>06-01-2000</td>
</tr>
</tbody>
</table>