

Bootstrapping in Lightweight Machine to Machine Systems

Technical field

- 5 The invention relates to a method of managing bootstrapping of a client of a utility metering device in a Lightweight Machine to Machine (LwM2M) system.

Background

- 10 The Open Mobile Alliance (OMA) Lightweight Machine to Machine (LwM2M) protocol is a protocol for the management and service enablement of Internet of Things (IoT) devices, such as utility metering devices. Utility metering devices may comprise electricity meters, water meters, gas meters etc.
- 15 The device management capabilities provided by the LwM2M standard include the remote provisioning of security credentials to enable a device to access a LwM2M server (bootstrapping), updating the firmware of the device, fault management, remote device diagnostics and troubleshooting, and connectivity management.
- 20 Figure 1 shows a LwM2M architectural diagram. A typical LwM2M system 100 comprises a LwM2M client 102, located on a device 104, which may be a utility metering device. The LwM2M client 102 may be integrated as a software library or a built-in function of the device 104.
- 25 The LwM2M client 102 communicates with one or more LwM2M server(s). To initiate and maintain communication between the LwM2M client 102 and the one or more LwM2M server(s), the following LwM2M interfaces are defined: bootstrapping; client registration; device management and service enablement; and information reporting.
- 30 The bootstrap interface defines the set of commands that an LwM2M bootstrap server 106 uses to provision the initial configuration of the LwM2M client 102 and allow subsequent registration and communication between the LwM2M client 102 and the LwM2M server(s) 108. The LwM2M client 102 communicates with the LwM2M bootstrap server 106 during its first, or alternatively every, boot-up. The LwM2M
- 35 bootstrap server 106 configures the LwM2M client 102 so that it is in an appropriate

state for communication with the one or more LwM2M servers 108. This configuration may comprise the transmission of bootstrap commands from the LwM2M bootstrap server 106 and to the LwM2M client 102, which causes objects and resources to be configured and/or deleted, created and/or written to the LwM2M client 102.

5

Each object 110a-c of the LwM2M client 102 is accessible by the one or more LwM2M servers 108, and represents a different concept of data. For example, a first object 110a may be for managing connections with the LwM2M servers 108, a second object 110b may be for providing a GPS location of the device 104, and a third object 110c may be for managing network connections. The objects 110a-c may be single-instance or multiple-instance. Each object may comprise an object identifier and information on the number of instances (i.e. single or multiple). Each object instance of an object 110a-c comprises at least one resource comprising an item of data. For example, a location object may comprise six resources: a latitude, a longitude, an altitude, a velocity, an uncertainty and a timestamp.

15

To allow the LwM2M client 102 to communicate with the LwM2M bootstrap server 106 on first boot-up of the LwM2M client 102, the LwM2M client 102 needs to be pre-injected with the LwM2M bootstrap server 106 details, such as a bootstrap identifier, for example a uniform resource identifier (URI). The URI may comprise a fully qualified domain name (FQDN), and/or a bootstrap server certificate. This may be achieved by injecting the LwM2M client 102 with a security object comprising the details of the LwM2M bootstrap server 106.

20

Generally, however, at the point of manufacture of devices such as metering devices, the end customer LwM2M bootstrap server details are not known, since it is not known which customer will eventually receive which device. As such, devices 104 will be pre-injected with details of a global bootstrap server. When the metering device, and therefore the LwM2M client 102, is booted up, the LwM2M client 102 will communicate with the global bootstrap server, and the global bootstrap server will configure the LwM2M client for registration and communication with the LwM2M server(s) 108.

25

30

There are a number of disadvantages of such methods however. For example, if there are any configuration errors when the LwM2M client 102 is bootstrapped, or if an LwM2M server 108 or client 102 certificate expires (such as an LwM2M registration

35

server certificate), the LwM2M client 102 will fail to register with the LwM2M server 108. The LwM2M client 102 will then re-initiate bootstrapping with the global bootstrap server. This results in significant scale challenges for the global bootstrap server, which may be required to handle bootstrapping of new devices across the globe, as
5 well as bootstrapping of known devices which experience configuration errors.

Furthermore, security and privacy issues may arise, due to the storage of customer data at the global bootstrap server.

10 There exists a need to provide an alternative method of bootstrapping devices in an LwM2M system.

Summary

15 According to the invention in a first aspect, there is provided a method of managing bootstrapping of a client of a utility metering device in a Lightweight Machine to Machine (LwM2M) system, wherein bootstrapping comprises configuring the client for connection and registration with one or more device management servers, the method comprising: configuring the client with a security object comprising a first security object
20 instance associated with a first bootstrap server, the first security object instance comprising data for use by the client in establishing communication with the first bootstrap server; configuring the client, by the first bootstrap server, with a final bootstrap security object instance associated with a final bootstrap server and comprising data for use by the client in establishing communication with the final
25 bootstrap server, the final bootstrap server for configuring the client for connection and registration with one or more device management servers; establishing, by the client and with the final bootstrap server, communication using the final bootstrap security object instance; and configuring, by the final bootstrap server, the client for connection and registration with the one or more device management servers, wherein if the client
30 is unable to connect and/or register with the one or more device management servers, the method further comprises initiating communication between the client and the final bootstrap server for reconfiguration of the client by the final bootstrap server for a subsequent connection and/or registration attempt with the one or more device management servers.

35

Optionally, configuring, by the final bootstrap server, the client for connection and registration with the one or more device management servers comprises: configuring the client with a device management security object instance associated with the one or more device management servers and comprising data for use by the client in communicating with the one or more device management servers.

Optionally, the reconfiguration of the client by the final bootstrap server comprises configuring the client with an updated device management security object instance.

Optionally, if the client is unable to connect and/or register with the one or more device management servers following the reconfiguration by the final bootstrap server, the method further comprises initiating communication between the client and the first bootstrap server for reconfiguration of the client by the first bootstrap server.

Optionally, the first bootstrap server configures clients of a plurality of utility metering devices, and wherein the final bootstrap server configures clients of a subset of the plurality of utility metering devices.

Optionally, the first bootstrap server configures the client with an intermediate bootstrap security object instance associated with an intermediate bootstrap server and comprising data for use by the client in communicating with the intermediate bootstrap server, and wherein the method further comprises: establishing, by the client and with the intermediate bootstrap server, data communication using the intermediate bootstrap security object instance; and configuring the client, by the intermediate bootstrap server, with the final security object instance associated with the final bootstrap server such that the client can establish data communication with the final bootstrap server for connection and registration with the one or more device management servers.

Optionally, if the client is unable to connect and/or register with the one or more device management servers following the reconfiguration of the client by the final bootstrap server, the method further comprises determining a sequence in which communication should be initiated with the final bootstrap server, the intermediate bootstrap server and the first bootstrap server for reconfiguration of the client by each of the final bootstrap server, the intermediate bootstrap server and the first bootstrap server.

Optionally, the client initiates communication for reconfiguration with one of the final bootstrap server, the intermediate bootstrap server and the first bootstrap server based on the determined sequence, and wherein communication with a next bootstrap server in the determined sequence is only initiated if, after reconfiguration by the one of the final bootstrap server, the intermediate bootstrap server and the first bootstrap server, registration with the one or more device management servers fails.

Optionally, if the client is unable to connect and/or register with the one or more device management servers following the reconfiguration of the client by the final bootstrap server, the method further comprises: initiating communication between the client and the intermediate bootstrap server for reconfiguration of the client by the intermediate bootstrap server; and wherein if the client is unable to connect and/or register with the one or more device management servers following the reconfiguration of the client by the intermediate bootstrap server, the method comprises initiating communication between the client and the first bootstrap server for reconfiguration of the client by the first bootstrap server.

Optionally, the first bootstrap server configures clients of a plurality of utility metering devices, the intermediate bootstrap server configures clients of a subset of the plurality of utility metering devices, and wherein the final bootstrap server configures clients of at least one of the subset of the plurality of metering devices.

Optionally, if the final bootstrap security object instance comprises data indicating that an enrolment over secure transport, EST, certificate is needed for registration with the one or more device management servers, the method further comprises: sending, by the client and to an EST server, an enrolment over secure transport, EST, certificate request; and receiving, by the client and from the EST server, the EST certificate; using the EST certificate for connection and/or registration with the one or more device management servers.

Optionally, the client comprises a first bootstrap server certificate for authenticating communication between the client and the first bootstrap server, and a final bootstrap server certificate for authenticating communication between the client and the final bootstrap server, and wherein the method comprises: determining, by the client, an

expiry date of the first bootstrap server certificate and/or the final bootstrap server certificate; and initiating replacement of the first bootstrap server certificate and/or the final bootstrap server certificate if the respective expiry dates fall within a threshold time period.

5

Optionally, the client further comprises an intermediate bootstrap server certificate for authenticating communication between the client and the intermediate bootstrap server, and wherein the method further comprises: determining, by the client, an expiry date of the intermediate bootstrap server certificate; and initiating replacement of the intermediate bootstrap server certificate if the expiry date falls within the threshold time period.

10

Optionally, the first bootstrap server is configured to configure the client on a first boot-up of the client.

15

According to the invention in a further aspect, there is provided a method of managing bootstrapping of a client of a utility metering device in a Lightweight Machine to Machine (LwM2M) system, wherein bootstrapping comprises configuring the client for connection and registration with one or more device management servers, the method comprising: configuring the client with a security object comprising a first security object instance associated with a first bootstrap server, the first security object instance comprising data for use by the client in establishing communication with the first bootstrap server; configuring the client, by the first bootstrap server, with an intermediate bootstrap security object instance associated with an intermediate bootstrap server and comprising data for use by the client in communicating with the intermediate bootstrap server, establishing, by the client and with the intermediate bootstrap server, data communication using the intermediate bootstrap security object instance; configuring the client, by the intermediate bootstrap server, with a final security object instance associated with the final bootstrap server and comprising data for connection and registration with the one or more device management servers; with a final bootstrap security object instance associated with a final bootstrap server and comprising data for use by the client in establishing communication with the final bootstrap server, the final bootstrap server for configuring the client for connection and registration with one or more device management servers; establishing, by the client and with the final bootstrap server, communication using the final bootstrap security

20

25

30

35

object instance; and configuring, by the final bootstrap server, the client for connection and registration with the one or more device management servers, wherein if the client is unable to connect and/or register with the one or more device management servers, the method further comprises initiating communication between the client and the final
5 bootstrap server for reconfiguration of the client by the final bootstrap server for a subsequent connection and/or registration attempt with the one or more device management servers.

Brief description of drawings

10

Figure 1 shows an exemplary Lightweight Machine to Machine (LwM2M) architecture;
Figure 2 shows an exemplary device comprising an LwM2M client;
Figure 3 shows a schematic view of an exemplary security object instance;
Figure 4 shows an exemplary sequence diagram; and
15 Figure 5 shows an exemplary sequence diagram.

Detailed description

Generally disclosed herein is a method of managing bootstrapping of a client of a utility
20 metering device in a Lightweight Machine to Machine (LwM2M) system. As used herein, "bootstrapping" refers to the process of configuring a client of a device for connection and registration with one or more servers, which may be device management servers. Bootstrapping may comprise provisioning the client with the information/data and credentials needed to allow the client to connect with the one or
25 more servers.

The method comprises injecting, or providing, the client of a utility metering device with an identifier (for example, a URI) associated with a first bootstrap server, such that the client initiates data communication with the first bootstrap server on the first boot-up of
30 the client. The first bootstrap server may be a global bootstrap server configured to communicate with a large number of utility metering device clients on their first boot-ups, regardless of who the intended end customer of the utility metering device is or of where the intended installation location is. Providing the client with the details of the first bootstrap server may comprise configuring the client with a security object
35 comprising a first security object instance associated with the first bootstrap server.

The first security object instance may comprise the identifier that allows the client to communicate with the first bootstrap server and/or a security credential that allows authentication of the communication between the client and the first bootstrap server.

5 In response to receiving the communication from the client on the first boot-up of the client, the first bootstrap server configures the client with a further security object instance associated with a further bootstrap server. The further bootstrap server may be a final bootstrap server for configuring the client for connection and registration with the one or more device management servers, and the further security object instance
10 may be a final bootstrap security object instance associated with the final bootstrap server. The final bootstrap server configures the client with a device management security object instance associated with the one or more device management servers. Following configuration of the client by the final bootstrap server, the client communicates with the one or more device management servers for connection and
15 registration using the device management security object instance.

Alternatively, the further bootstrap server may be associated with an intermediate bootstrap server. The intermediate bootstrap server may, for example, be a regional bootstrap server for configuring a subset of the devices served by the first (e.g. global)
20 bootstrap server. The subset of devices served by the intermediate bootstrap server may be determined based on location and/or intended end customer. In this example, the first bootstrap server configures the client with an intermediate bootstrap security object instance associated with the intermediate bootstrap server, and the client then communicates with the intermediate bootstrap server. The intermediate bootstrap
25 server configures the client with the final bootstrap security object instance associated with the final bootstrap server. This allows the client to communicate with the final bootstrap server for configuration of the client for connection and registration with the one or more device management servers. That is, the final bootstrap server configures the client with the device management security object instance associated with the one
30 or more device management servers for connection and registration of the client thereto using the device management security object instance. The skilled person will appreciate that there may be one or more intermediate bootstrap servers that the client communicates with in sequence before the final bootstrap server and a similar process will apply.

35

The client is configured such that if the client is unable to connect and/or register with the one or more device management servers, which may occur if a certificate of the one or more device management servers expires for example, the method further comprises initiating communication between the client and the final bootstrap server for reconfiguration of the client by the final bootstrap server for a subsequent connection and/or registration attempt. If this reconfiguration is unsuccessful (i.e. the client is still unable to connect and/or register with the one or more device management servers), the method may further comprise initiating communication between the client and the first bootstrap server, or if there are intermediate bootstrap servers, with the intermediate bootstrap servers for reconfiguration of the client.

Advantageously, therefore, in the method proposed by the invention, communication between the client and the first bootstrap server (which serves the largest number of clients) is avoided unless necessary. The client always firstly tries for reconfiguration at the final or intermediate bootstrap servers, and it is only if these attempts are also unsuccessful that communication is initiated with the first bootstrap server. This reduces the load placed on the first bootstrap server and therefore ameliorates scaling and load issues associated with the first bootstrap server.

Figure 2 shows a schematic representation of a device 104 for use in a LwM2M communications system. The device 104 may be a utility metering device, such as a gas, water or electricity metering device. The device 104 may comprise a LwM2M client 102 (referred to hereafter as a client) integrated as a software library or a built-in function of the device 104. The LwM2M client may communicate with one or more bootstrap servers and one or more device management servers, as will be outlined in more detail below.

The device 104 may comprise a communications module 112. The communications module 112 may be configured to facilitate communications with the one or more bootstrap servers and one or more device management servers. The communications module 112 may use wireless communication such as wireless local area network (Wi-Fi) communications, Bluetooth®, cellular communications such as 3G or 4G, or radio frequency (RF) communications. This skilled person will appreciate that these communications techniques are exemplary only and other communications may be used in alternative arrangements. The device 104 may use any suitable protocols for

communicating with the one or more bootstrap servers and the one or more device management servers. For example, Constrained Application Protocol (CoAP). In particular, the device 104 may communicate with the one or more bootstrap servers and the one or more device management servers using protocols compliant with the
5 OMA LWM2M specification.

The device 104 may further comprise a memory 114 and a processor 116. The memory 114 may comprise a non-volatile memory and/or a volatile memory. The memory 114 may have a computer program 118 stored therein. The computer
10 program 118 may comprise instructions for performing the methods disclosed herein. The computer program 118 may be loaded in the memory 114 from a non-transitory computer readable medium 120, on which the computer program is stored. The processor 116 is configured by the computer program 118 to perform one or more of the functions of the client 102.

15 The memory 114 may also be configured to store credential data, such as a certificate, security key or other identifier. The credential data may be used by the device 104 to authenticate with and allow connection to and/or registration with one or more bootstrap servers and/or data management servers.

20 Each of the client 102, communications module 112, memory 114, processor 116 is in data communication with the other components 102, 112, 114, 116 of the device 104. The device 104 can be implemented as a combination of computer hardware and software. The memory 114 stores the various programs/executable files that are implemented by a processor 116, and also provides a storage unit for any required
25 data.

As outlined above, Figure 1 shows an example architecture which illustrates the relationship between a device 104 comprising a client 102, such as an LwM2M client,
30 and one or more servers, which comprise LwM2M bootstrap server(s) 106, referred to as bootstrap servers hereafter, and LwM2M device management server(s) 108, referred to as device management servers hereafter.

The client 102 may comprise a security object. The security object may comprise the
35 details and credentials for the client 102 to communicate and/or register with bootstrap

servers and/or device management servers. The security object may comprise one or more security object instances, wherein each security object instance is associated with a respective server (e.g. a bootstrap server or a device management server). The bootstrap servers disclosed herein are each configured to provision (or configure) the client with one or more security object instances. Each security object instance may comprise a plurality of resources, wherein each of the resources comprises an item of data.

Figure 3 shows an exemplary security object instance 300, which may be provisioned to the client 102 by a bootstrap server.

The security object instance 300 may comprise an object identifier 302. In this case, the object identifier indicates that the security object instance is associated with a security object. The object identifier 302 may comprise a single number. Conventionally, LwM2M security objects are assigned an identifier of "0".

The security object instance 300 may comprise an object instance identifier 304. The first security object instance may be provided with an identifier of "0" (as shown in Figure 3). As will be described in more detail below, subsequent security object instances will be assigned identifiers that increase numerically, i.e. the next security object instance will be assigned an identifier of 1, the next an identifier of 2 and so on.

The security object instance 300 may comprise a plurality of resources 306a-n, each comprising an item of data. The plurality of resources 306a-n may each comprise a resource identifier 308. The first resource 306a of the security object instance 300 may be provided with an identifier of "0" (as shown in Figure 3). Subsequent resources 306b-306n may be assigned identifiers that increase numerically, i.e. the next resource may be assigned an identifier of 1, the next an identifier of 2 and so on.

The objects, object instances and resources may be accessed using a URI, for example in the form: /{Object ID}/{Object Instance}/{Resource ID}.

Each security object instance may be associated with a bootstrap server or a device management server. That is, each security object instance may comprise data (in the form of resources) for facilitating communication between the client 102 and the

bootstrap server or device management server that the security object instance is associated with. The data for facilitating the communication is stored within the resources 306a-n of the security object instance 300.

- 5 The exemplary security object instance 300 comprises a server identifier resource, or server URI, 306a. The server identifier resource may comprise the URI, or another identifier, of a bootstrap server or a device management server, to allow the client 102 to communicate with the bootstrap server or the device management server.
- 10 The exemplary security object instance 300 may comprise a resource 306b comprising an indication of whether the server associated with the security object instance 300 is a bootstrap server or not. The indication may comprise a true/false indication, wherein if the associated server is a bootstrap server, the resource 306b indicates "true" and if the associated server is not a bootstrap server (e.g. if it is a device management
- 15 server) the resource 306b indicates "false".

The exemplary security object instance 300 may comprise a security mode resource 306c indicating which security mode should be used when establishing communications between the client 102 and the server associated with the security

20 object instance 300. For example, the security mode resource 306c may indicate that one of a pre-shared key mode, a raw public key mode, a certificate mode, a certificate mode with EST or no certificate mode should be used. The skilled person will appreciate that alternative security modes to those listed herein may also be used, and the above list is exemplary only.

- 25 At least some of the resources 306a-n may be populated in dependence on the security mode specified within the security mode resource 306c. For example, in the pre-shared key mode, communication is encrypted and authenticated using a secret key shared between the client 102 and the server. As such, for this mode a secret key
- 30 may be stored in the secret key resource 306f. In the certificate mode, certificates are generated for both the client 102 and the server associated with the security object instance 300, and the certificates are stored in client certificate and server certificate resources 306d, 306e respectively. A secret key resource 306f comprises the client private key. The client and server certificates and the secret key may be used to
- 35 authenticate connection and communication between the client 102 and the server

associated with the security object instance 300. The skilled person will appreciate that in different security modes, the data stored within the resources 306a-n may differ in dependence on the specified security mode.

5 The exemplary security object instance 300 may comprise a short server identifier resource 306n. The short server identifier resource 306n uniquely identifies each device management server configured for the client 102. The short server identifier resource 306n is populated when the bootstrap server identifier resource 306b indicates that the associated server is not a bootstrap server (i.e. recites “false”).

10

The skilled person will understand that the security object instance shown in Figure 3 is exemplary only, and that further and/or different resources may be contained within alternative security object instances.

15 An exemplary method of managing bootstrapping of a client of a utility metering device in a Lightweight Machine to Machine (LwM2M) system will now be described with reference to Figure 4.

20 400: After, or during, manufacture of a device 104, which comprises a client 102, and which may be a utility metering device, the device 104 is provided with a first bootstrap server identifier. The first bootstrap server identifier is configured to allow the client 102 to communicate with a first bootstrap server 106a and may comprise a URI, or any other identifier to allow the device 104 to communicate with a first bootstrap server 106a.

25

The first bootstrap server 106a may comprise a global bootstrap server configured to provision clients of a plurality of devices. The plurality of devices may comprise a large number of devices, and in some instances, may comprise all devices manufactured and provided by a single company regardless of the end customer or installation location. In alternative arrangements, the plurality of devices may include devices manufactured by multiple companies.

30

Providing the device 104 with the first bootstrap server identifier may comprise provisioning the client 102 with a security object comprising a first security object instance associated with the first bootstrap server 106a. The first

35

security object instance may be the security object instance 300 and comprise one or more of an object ID 302, an object instance ID 304 and a plurality of resources 306a-n with associated resource IDs 308.

5 In this example, the object ID 302 is "0", which as outlined above, is conventional for LwM2M security objects. The first object instance ID 304 is "0".

10 The first security object instance may comprise resources 306a-n comprising data for use by the client 102 to establish communication with the first bootstrap server 106a. For example, the first security object instance may comprise one or more of: a server identifier resource 306a comprising the URI of the first bootstrap server 106a; a bootstrap server indication resource 306b indicating that the first bootstrap server 106a is a bootstrap server; a security mode resource 306c indicating which security mode should be used to authenticate communications (e.g. a certificate mode or a certificate mode with EST etc.); client and server certificate resources 306d, 306e comprising the client and first bootstrap server certificates respectively, and a secret key resource 306f comprising the secret key.

20 402: Using the first security object instance, which as explained above is associated with the first bootstrap server 106a, the client 102 sends a request to the first bootstrap server 106a for bootstrapping. In response to the request, the first bootstrap server 106a bootstraps the client 102.

25 Bootstrapping the client 102 comprises the first bootstrap server 106a configuring the client 102 with a further security object instance, which may be substantially the same in structure as the exemplary security object instance shown in Figure 3.

30 In the particular example shown in Figure 4, the further security object instance is a final bootstrap security object instance associated with a final bootstrap server 106b. However as will be explained in more detail below, this may not be the case in all methods, and alternative methods may involve the first bootstrap server configuring the client 102 with a further security object instance

35

associated with an intermediate bootstrap server (which in turn, configures the client with the final bootstrap security object instance for communication with the final bootstrap server 106b).

- 5 In this example, the final bootstrap security object instance may comprise an object instance identifier 304 of "1". The identifiers assigned to each new security object instance configured during bootstrapping of the client may increase numerically by one.
- 10 The final bootstrap security object instance may comprise resources 306a-n comprising data for use by the client 102 to establish communication with the final bootstrap server 106b. For example, the final bootstrap security object instance may comprise one or more of: a server identifier resource 306a comprising the URI of the final bootstrap server 106b; a bootstrap server
- 15 indication resource 306b indicating that the final bootstrap server is a bootstrap server (i.e. indicating a "true" condition); a security mode resource 306c indicating which security mode should be used to authenticate communications (e.g. a certificate mode or a certificate with EST mode etc.); client and server
- 20 certificate resources 306d, 306e comprising the client and final bootstrap server certificates respectively, and a secret key resource 306f comprising the secret key.
- 404: Using the final bootstrap security object instance, which as explained above is associated with the final bootstrap server 106b, the client 102 sends a request
- 25 to the final bootstrap server 106b for bootstrapping. In response to the request, the final bootstrap server 106b bootstraps the client 102.
- As used herein, the term "final bootstrap server" means the bootstrap server configured to provision (or configure) the client 102 for communication with one
- 30 or more device management servers 108.
- The final bootstrap server 106b may configure a subset of the devices configured by the first bootstrap server 106a. For example, the first bootstrap server 106a may be a global bootstrap server for configuring all devices from a
- 35 particular manufacturer, while the final bootstrap server 106b may be

configured to provision a subset of those devices based on, for example, on which end customer the devices are used by. As such, the final bootstrap server 106b is responsible for configuring a smaller number of devices than the first bootstrap server 106a.

5

The final bootstrap server 106 configures the client 102 to enable the client 102 to register and connect with one or more device management servers 108. Configuring the client 102 for registration and connection with the one or more device management servers 108 comprises configuring the client with a device management security object instance associated with the one or more device management servers 108. The device management security object instance may be substantially the same in structure as the exemplary security object instance shown in Figure 3. In this example, the device management security object instance may comprise a security object instance ID 304 of "2".

10

15

The device management security object instance may comprise resources 306a-n comprising data for use by the client to establish communication with the one or more device management servers 108. For example, the device management security object instance may comprise one or more of: a server identifier resource 306 comprising the URI of the one or more device management servers 108; a bootstrap server indication resource 306b indicating that the one or more device management servers 108 are not bootstrap servers (i.e. indicating a "false" condition); a security mode resource 306c indicating that which security mode should be used to authenticate communications (e.g. certificate mode or certificate with EST mode); client and server certificate resources 306d, 306e comprising the client and the one or more device management server certificates respectively; and a secret key resource 306f comprising the secret key. The device management security object instance may further comprise a short server ID resource 306n comprising identifiers for each of the one or more device management servers configured for the client 102.

20

25

30

35

In exemplary arrangements, the final bootstrap security object instance may comprise a security mode resource 306c indicating that a certificate mode with Enrolment over Secure Transport, EST, should be used. If the security mode is

- a certificate mode with EST, after the final bootstrap server 106b has finished bootstrapping the client 102, the client 102 is configured to initiate communication with an EST server and send an EST certificate enrolment request to the EST server. In response to the EST certificate enrolment request, the EST server may send the client 102 an EST certificate for authentication of the client 102 and the device management servers 108. The EST certificate may be stored and used by the client 102 to register and connect with the device management servers 108.
- 5
- 10 Bootstrapping by the final bootstrap server 106b may further comprise configuring the client with a server object instance comprising resources comprising data relating to the one or more device management servers 108. The server object instance may comprise a server identifier resource identifying the one or more device management servers 108 and a short server ID of the
- 15 the one or more device management servers 108, similar to the device management security object instance. The server object instance may further comprise a lifetime resource and/or a notification storing when disabled or offline recourse. The skilled person will appreciate that the server object instance may comprise further resources. The server object instance and the
- 20 device management security object instance may both be used by the client 102 to connect and register to the one or more device management servers 108.
- 25 406: Using the device management security object instance, which as explained above is associated with the one or more device management servers 108, the client 102 sends a registration request to the one or more device management servers 108.
- 30 If the client 102 successfully registers with the one or more device management servers 108, the client 102 is able to receive commands from the one or more device management servers 108 on the device management and service enablement interface and the information reporting interface, and therefore assist in managing the device 104.

In some instances however, the registration process fails. The registration process may fail, for example, if one or more of the client and device management server certificates have expired, or because the device 104 has not yet been registered with the one or more device management servers 108 as a valid device.

5

408: If the registration process fails, or the client 102 loses connection with the one or more device management servers 108 following registration, the client 102 may be configured to initiate communication with one of the bootstrap servers 106a, 106 for reconfiguration.

10

The client 102 is configured to determine which bootstrap server 106a, 106b communication should be initiated with for reconfiguration of the client 102. In exemplary arrangements, the client 102 may comprise a reconfiguration determiner configured to determine which bootstrap server 106a, 106b should reconfigure the client 102 if registration or connection of the client 102 with the one or more device management servers 108 fails. The client 102 is able to initiate communications with either the first bootstrap server 106a or the final bootstrap server 106b for reconfiguration because the client comprises security object instances associated with each of the first bootstrap server 106a and the final bootstrap server 106b, as outlined above. This is in contrast to prior art arrangements, in which the client comprises only a single security object instance associated with the first bootstrap server. As such, in prior art arrangements, if the registration process fails, the client is only able to initiate communications with the first bootstrap server 106a, which as outlined above, typically serves significantly more devices than the final bootstrap server 106b, leading to load and scaling issues at the first bootstrap server 106a.

15

20

25

In exemplary methods, the reconfiguration determiner determines which bootstrap server 106a, 106b should reconfigure the client 102 based on a pre-defined hierarchy or sequence. For example, the pre-defined hierarchy or sequence may define that if registration of the client 102 with the one or more device management servers 108 fails, then the client 102 should firstly contact the final bootstrap server 106b for reconfiguration. The hierarchy or sequence may further define that if registration of the client 102 with the one or more

30

35

device management servers 108 fails after reconfiguration of the client 102 by the final bootstrap server 106b, the client 102 should contact the first bootstrap server 106b for reconfiguration.

5 The hierarchy or sequence may be defined based on the security object instance identifiers 304 associated with the servers 106a, 106b, 108. Specifically, the hierarchy or sequence may specify that the servers 106a, 106b should be contacted in a numerical sequence based on the security object instance identifiers 304 of the security object instances associated with the
10 servers 106a, 106b respectively. In this example, as outlined above, the final bootstrap security object instance associated with the final bootstrap server 106b has a security object instance identifier of "1", and the first security object instance associated with the first bootstrap server 106a has a security object instance identifier of "0". The hierarchy or sequence may define that the
15 bootstrap servers 106a, 106b should be contacted based on the security object instance identifiers 304 in reverse numerical order. As such, the final bootstrap server 106b is contacted for reconfiguration before the first bootstrap server 106a, and the first bootstrap server 106a is only contacted if reconfiguration by the final bootstrap server 106b does not result in a successful registration attempt with the one or more device management servers 108.
20

 The client 102 may be configured to initiate communication with one of the bootstrap servers 106a, 106b for reconfiguration, if the number of attempts of the client 102 to register with the one or more device management servers 108 exceeds a threshold. This ensures that the client 102 does not communicate with the one of the bootstrap servers 106a, 106b for reconfiguration in the event of an anomalous failure to register or maintain connection with the device management servers 108, or an unrelated failure (such as a server failure), which reconfiguration would not fix. The threshold may be set at substantially
25 any number based on the system and/or customer requirements.
30

 In exemplary methods, if the number of attempts of the client 102 to register with the one or more device management servers 108 exceeds the threshold, the client 102 may be further configured to only initiate communication with one of the bootstrap servers 106a, 106b for reconfiguration upon expiry of
35

predefined backoff period. The predefined backoff period may be, for example, 1 hour, 2 hours, 10 hours, 24 hours. Advantageously, this allows time for the one or more of the bootstrap servers 106a, 106b to be updated to provision the client 102 with updated security object instances on receipt of a reconfiguration request from the client 102, in the event the registration failure is caused by certificate expiry, or else time for the device 104 to be registered with the one or more device management servers 108 as a valid device.

410: The client 102 initiates communication with the determined bootstrap sever for reconfiguration, for a subsequent attempt at registration/connection with the one or more device management servers 108.

In exemplary methods, the client 102 may initiate communication with the determined bootstrap server if the number of attempts of the client 102 to register with the one or more device management servers 108 exceeds the threshold, and optionally the back-off period expires.

In this example, the determined bootstrap server is the final bootstrap server 106b.

In response to the reconfiguration request received from the client 102, the final bootstrap server 106b reconfigures the client 102. Reconfiguring may comprise updating the device management security object instance associated with the one or more device management servers 108 and/or updating the server object associated with the one or more device management servers 108. For example, one or more resources of the device management security object instance and/or the server object associated with the one or more device management servers 108 may be updated. This may occur, for example, if one or more of the previous client or server certificates have expired, and therefore new client and/or server certificates now apply. In such a case, the client and/or server resources 306d, 306e of the device management security object instance would be updated. The skilled person will understand that in alternative cases one or more of the other resources may be updated (or alternatively no resources may be updated if there have been no configuration changes).

- 412: Following the reconfiguration of the client 102 by the final bootstrap server 106b, the client 102 attempts registration and/or connection with the one or more device management servers 108. If the registration and connection attempt is successful, no further action is taken and the client 102 communicates with the one or more device management servers 108 on the device management and service enablement interface and the information reporting interface.
- 414: If the client 102 is unable to register and/or connect with the one or more device management servers 108 following reconfiguration by the final bootstrap server 106b, the client 102 may initiate communication with the next bootstrap server in the hierarchy or sequence. The client 102 may initiate communication with the next bootstrap server in the hierarchy or sequence if the number of attempts of the client 102 to register with the one or more device management servers 108 exceeds the threshold, and optionally the back-off period expires.
- In this example, the next bootstrap server in the hierarchy or sequence is the first bootstrap sever 106a.
- In response to the reconfiguration request received from the client 102, the first bootstrap server 106a reconfigures the client 102. Reconfiguring may comprise updating the final bootstrap security object instance associated with the final bootstrap server 106b. For example, one or more resources of the final bootstrap security object instance may be updated. This may occur, for example, if one or more of the previous client or server certificates have expired, and therefore new client and/or server certificates now apply. In such a case, the client and/or server resources 306d, 306e of the final bootstrap security object instance would be updated. Alternatively, a system update may have resulted in a different final bootstrap server being responsible for configuring the client 102 for registration with the one or more device management servers 108. In such a case, the final bootstrap security object instance may be reconfigured with the resources/data associated with the different final bootstrap server. The client 102 may then attempt communication with the final bootstrap server associated with the updated final bootstrap

security object instance (which may be the final bootstrap server 106b or a different final bootstrap server) for bootstrapping by the final bootstrap server for a subsequent registration attempt with the one or more device management servers 108.

5

The skilled person will appreciate that the above-described method advantageously ensures that not all requests for reconfiguration are directed towards the first bootstrap server 106a in the first instance. As mentioned above, typically the first bootstrap server 106a is responsible for a larger number of devices than the final bootstrap server (which typically is responsible for a subset of the devices served by the first bootstrap server). By ensuring that the final bootstrap server 106b, which is responsible for the smaller number of devices, reconfigured the client 102 in the first instance, the load placed on the first bootstrap server 106a is reduced.

10

15

The skilled person will also appreciate that the client 102 may still be unable to register and/or connect with the one or more device management servers 108 following the reconfiguration of the client 102 by the first bootstrap server 106a in step 414. This may be as a result of a failure or error unrelated to the certificates or configurations, for example a server failure. In such instances, the client 102 may be configured to re-attempt registration after a predefined back-off period. Alternatively, or additionally, the client 102 may be configured to send an alert to a third party device that registration has failed. This may allow a developer to look further into the problem and fix the server error, for example.

20

25

The skilled person will appreciate that in alternative arrangements further bootstrap servers may be used. For example, in some arrangements, the first bootstrap server may provision a plurality of devices for communication with an intermediate bootstrap server, which provisions a subset of the plurality of devices for communication with the final bootstrap server. The final bootstrap server then provisions one or more of the devices provisioned by the intermediate bootstrap server for registration and connection with the one or more device management servers.

30

An exemplary method of managing bootstrapping of a client of a utility metering device in a Lightweight Machine to Machine (LwM2M) system comprising an intermediate bootstrap server 106c will now be described with reference to Figure 5.

35

- 500: The same process outlined above in respect of step 400 (Figure 4) is followed here and not repeated in detail. The client 102 is provided with a security object comprising a first security object instance associated with the first bootstrap server 106a. The first security object instance may comprise an object instance identifier 304 of "0". The features outlined in step 400 in respect of the first bootstrap server 106a and the first security object instance apply here.
- 502: Using the first security object instance, which as explained above is associated with the first bootstrap server 106a, the client 102 sends a request to the first bootstrap server 106a for bootstrapping. In response to the request, the first bootstrap server 106a bootstraps the client 102.
- In this particular example, bootstrapping the client 102 comprises the first bootstrap server 106a configuring the client with an intermediate bootstrap security object instance associated with the intermediate bootstrap server 106c. The intermediate bootstrap security object instance may comprise an object instance identifier 304 of "1". The intermediate bootstrap security object instance may be substantially the same in structure as the exemplary security object instance shown in Figure 3.
- The intermediate bootstrap security object instance may comprise resources 306a-n comprising data for use by the client 102 to establish communication with the intermediate bootstrap server 106c. For example, the intermediate bootstrap security object instance may comprise one or more of: a server identifier resource 306a comprising the URI of the intermediate bootstrap server 106c; a bootstrap server indication resource 306b indicating that the intermediate bootstrap server 106c is a bootstrap server (i.e. indicating a "true" condition); a security mode resource 306c indicating which security mode should be used to authenticate communications (e.g. a certificate mode or certificate with EST mode); client and server certificate resources 306d, 306e comprising the client and intermediate bootstrap server certificates respectively, and a secret key resource 306f comprising the secret key.

504: Using the intermediate bootstrap security object instance, which as explained above is associated with the intermediate bootstrap server 106c, the client 102 sends a request to the intermediate bootstrap server 106c for bootstrapping. In response to the request, the intermediate bootstrap server 106c bootstraps the client 102.

The intermediate bootstrap server 106c may configure a subset of the devices configured by the first bootstrap server 106a. For example, the first bootstrap server 106a may be a global bootstrap server for configuring all devices from a particular manufacturer, while the intermediate bootstrap server 106c may be a regional bootstrap server configured to provision a subset of the devices based on, for example, the geographical region that the device is to be installed in.

The intermediate bootstrap server 106c may configure the client 102 with the final bootstrap security object instance. The final bootstrap security object instance may comprise resources 306a-n comprising data for use by the client 102 to establish communication with the final bootstrap server 106b, as explained in step 402 above. The features outlined in step 402 in respect of the final bootstrap server 106b and the final bootstrap security object instance apply here, and are not repeated. In this example however, the final bootstrap security object instance may comprise an object instance identifier 304 of "2".

506: Using the final bootstrap security object instance, which as explained above is associated with the final bootstrap server 106b, the client 102 sends a request to the final bootstrap server 106b for bootstrapping. In response to the request, the final bootstrap server 106b bootstraps the client 102.

The final bootstrap server 106b may configure a subset of the devices configured by the intermediate bootstrap server 106c. For example, the final bootstrap server 106b may be a customer bootstrap server for configuring all devices managed by a particular customer.

As described above in respect of step 404 (Figure 4), the final bootstrap server 106b configures the client 102 to enable the client 102 to register and connect with one or more device management servers 108. Configuring the client 102

for registration and connection with one or more device management servers 108 comprises configuring the client with a device management security object instance associated with the one or more device management servers 108.

5 The device management security object instance may comprise resources 306a-n comprising data for use by the client 102 to establish communication with the final bootstrap server 106b, as explained in step 404 above. The features outlined in step 404 in respect of the device management servers and the device management security object instance apply here, and are not
10 repeated. In this example however, the device management security object instance may comprise an object instance ID 304 of "3".

 The bootstrapping by the final bootstrap server 106b may further comprise configuring the client with a server object instance, as described above in
15 respect of step 404 (Figure 4).

508: Using the device management security object instance, which as explained above is associated with the one or more device management servers 108, the client 102 sends a registration request to the one or more device management
20 servers 108.

 If the client 102 successfully registers with the one or more device management servers 108, the client 102 is able to receive commands from the one or more device management servers 108 on the device management and service
25 enablement interface and the information reporting interface, and therefore assist in managing the device 104.

510: If the registration process fails, or the client 102 loses connection with the one or more device management servers 108 following registration, the client 102
30 may be configured to determine which bootstrap server 106a, 106b, 106c communications should be initiated with for reconfiguration of the client 102.

 The determination of which bootstrap server 106a, 106b, 106c communications should be initiated with for reconfiguration of the client 102 is similar to the
35 process outlined above in respect of step 408 (and therefore the features

outlined in that passage apply here). That is, each bootstrap server 106a, 106b, 106c is contacted for reconfiguration in sequence, and the next bootstrap server in the sequence is only contacted if, after reconfiguration, registration with the one or more device management servers 108 fails.

5

Similarly to the process outlined in respect of step 408, the reconfiguration determiner determines which bootstrap server 106a, 106b, 106c should reconfigure the client based on a pre-defined hierarchy or sequence. The only difference is that in this example, there is an additional bootstrap server, the intermediate bootstrap server 106c, defined within the hierarchy or sequence. In this example, as outlined above, the final bootstrap security object instance associated with the final bootstrap server 106b has a security object instance identifier of "2", the intermediate security object instance associated with the intermediate bootstrap server 106c has a security object instance identifier of "1", and first security object instance associated with the first bootstrap server 106a has a security object instance identifier of "0".

10

15

As such, where the hierarchy or sequence defines that the bootstrap servers should be contacted in a reverse numerical sequence based on the security object instance identifiers 304, the final bootstrap server 106b would be contacted for reconfiguration first. If, after reconfiguration by the final bootstrap server 106b, registration with the one or more device management servers 108 fails, the intermediate bootstrap server 106c would be contacted. If after reconfiguration by the intermediate bootstrap server 106c, connection with the final bootstrap server 106b fails, the first bootstrap server 106a would be contacted. Therefore, the final bootstrap server 106b and the intermediate bootstrap server 106c are contacted for reconfiguration before the first bootstrap server 106a.

20

25

30

512: The client 102 initiates communication with the determined bootstrap sever for reconfiguration, for a subsequent attempt at registration/connection with the one or more device management servers 108.

35

In exemplary methods, the client 102 may initiate communication with the determined bootstrap server if the number of attempts of the client 102 to

register with the one or more device management servers 108 exceeds the threshold, and optionally the back-off period expires.

5 In this example, the determined bootstrap server is the final bootstrap server 106b. In response to the reconfiguration request received from the client 102, the final bootstrap server 106b reconfigures the client 102. Reconfiguring may comprise updating the device management security object instance associated with the one or more device management servers 108 and/or updating the server object associated with the one or more device management servers 108. 10 For example, one or more resources of the device management security object instance may be updated. This may occur, for example, if one or more of the previous client or server certificates have expired, and therefore new client and/or server certificates now apply. In such a case, the client and/or server resources 306d, 306e of the device management security object instance would be updated. 15

514: Following the reconfiguration of the client 102 by the final bootstrap server 106b, the client 102 attempts registration and/or connection with the one or more device management servers 108. If the registration and connection attempt is successful, no further action is taken and the client 102 communicates with the one or more device management servers 108 on the device management and service enablement interface and the information reporting interface. 20

25 516: If the client 102 is unable to register and/or connect with the one or more device management servers 108 following reconfiguration by the final bootstrap server 106b, the client 102 may initiate communication with the next bootstrap server in the hierarchy or sequence, as determined by the reconfiguration determiner. The client 102 may initiate communication with the next bootstrap server in the hierarchy or sequence if the number of attempts of the client 102 to register with the one or more device management servers 108 exceeds the threshold, and optionally the back-off period expires. 30

In this example, the next bootstrap server in the hierarchy or sequence is the intermediate bootstrap sever 106c. 35

5 In response to the reconfiguration request received from the client 102, the intermediate bootstrap server 106c reconfigures the client 102. Reconfiguring may comprise updating the final bootstrap security object instance associated with the final bootstrap server 106b.

10 518: Following the reconfiguration of the client 102 by the intermediate bootstrap server 106c, the client 102 attempts to connect with the final bootstrap server 106b for bootstrapping for a subsequent attempt at registration and/or connection with the one or more device management servers 108. If the connection attempt with the final bootstrap server 106b is successful, no further action is taken and the client 102 communicates with the final bootstrap server 106b for bootstrapping for registration and/or connection with the one or more device management servers 108.

15 520: If the client 102 is unable to connect with the final bootstrap server 106b following reconfiguration by the intermediate bootstrap server 106c, the client 102 may initiate communication with the next bootstrap server in the hierarchy or sequence. The client 102 may initiate communication with the next bootstrap server in the hierarchy or sequence if the number of attempts of the client 102 to connect with the final bootstrap server 106b exceeds the threshold, and optionally the back-off period expires.

20 In this example, the next bootstrap server in the hierarchy or sequence is the first bootstrap sever 106a.

25 In response to the reconfiguration request received from the client 102, the first bootstrap server 106a reconfigures the client 102. Reconfiguring may comprise updating the intermediate bootstrap security object instance associated with the intermediate bootstrap server 106c. For example, one or more resources of the intermediate bootstrap security object instance may be updated.

30 After reconfiguration, the client 102 may attempt connection with the intermediate bootstrap server 106c for bootstrapping by the intermediate bootstrap server 106 (and subsequent communication with the final bootstrap

server 106b, and registration with the one or more device management servers 108).

5 The skilled person will appreciate that the above-described method advantageously ensures that not all requests for reconfiguration are directed towards the first bootstrap server 106a or the intermediate bootstrap server 106c (which both typically serve a larger number of devices than the final bootstrap server 106b) in the first instance.

10 The skilled person will appreciate that in exemplary arrangements, each bootstrap sever 106a, 106b, 106c is configured to only reconfigure pre-specified security object instance(s), without altering (e.g. updating, deleting or overwriting) the other security object instances provisioned to the client 102. The pre-specified security object instances may comprise a security object instances associated with a single bootstrap server, or alternatively security object instances associated with one or more bootstrap
15 servers (such that a single bootstrap server reconfigures security object instances associated with one or more bootstrap servers).

In a particular example, each bootstrap server 106a, 106b, 106c may be configured to provision the client 102 with data relating to a single bootstrap server and/or the one or
20 more device management servers 108. Each bootstrap server 106a, 106b, 106c may therefore only configure or reconfigure the security object instances associated with the specified single bootstrap server and/or the one or more device management servers 108 without altering the security object instances associated with the remaining bootstrap server(s) and/or one or more device management servers 108.

25 In the above example, the final bootstrap server 106b is configured to provision the client with data to allow connection and registration with the one or more device management servers 108. As such, the final bootstrap server 106b is configured to reconfigure the device management security object instance associated with the one or
30 more device management servers 108, and to not reconfigure or otherwise alter the first security object instance associated with the first bootstrap server 106b, the intermediate bootstrap security object instance associated with the intermediate bootstrap server 106c, or the final bootstrap security object instance associated with the final bootstrap server 106b. Similarly, the intermediate bootstrap server 106c is
35 configured to provision the client 102 with data to allow connection with the final

bootstrap server 106. As such, the intermediate bootstrap server 106c is configured to reconfigure the final bootstrap security object instance associated with the final bootstrap server 106b, and does not reconfigure or otherwise alter the first security object instance associated with the first bootstrap server 106b, the intermediate
5 bootstrap security object instance associated with the intermediate bootstrap server 106c, or the device management security object instance associated with the one or more device management servers 108. Similarly, the first bootstrap server 106a is configured to provision the client 102 with data to allow connection with the intermediate bootstrap server 106c. As such, the first bootstrap server 106a is
10 configured to reconfigure the intermediate bootstrap security object instance associated with the intermediate bootstrap server 106c, and does not reconfigure or otherwise alter the first security object instance associated with the first bootstrap server 106b, the final bootstrap security object instance associated with the final bootstrap server 106b, or the device management security object instance associated
15 with the one or more device management servers 108.

As mentioned above, in alternative arrangements, at least one of the bootstrap servers 106a, 106b, 106c may be configured to reconfigure security object instances associated with one or more bootstrap servers and/or the one or more device
20 management servers 108. For example, the first bootstrap server 106a may be configured to update both the first security object instance and the intermediate bootstrap security object instance (or the final bootstrap security object instance where there is no intermediate bootstrap server 106c) if required. This ensures that if the certificate of the first bootstrap server 106a is about to expire, for example, the first
25 bootstrap server 106a is able to update/reconfigure the first security object instance with the new certificate details as well as reconfiguring the intermediate or final bootstrap security object instance.

The inventors have also appreciated that in order for the client 102 to take advantage
30 of the full reconfiguration hierarchy or sequence – i.e. for the client 102 to be able to in sequence contact the final bootstrap server 106b, the intermediate bootstrap server 106c and the first bootstrap server 106a if needed – the certificates associated with each bootstrap server 106a, 106b, 106c should still be live. If for example, the final bootstrap server 106b has a certificate that is expired, or set to expire before the
35 certificate of the intermediate bootstrap server 106c, then the client would not be able

to successfully communicate with the final bootstrap server 106b for reconfiguration, and therefore may not be able to successfully initiate reconfiguration or the reconfiguration sequence.

5 As such, in exemplary methods, the client 102 comprises a certificate expiry checker. The certificate expiry checker is configured to assess the respective bootstrap server certificate expiry date using, for example the data stored in the server certificate resource 306d which comprises the date and time that the respective server certificate is due to expire. If a respective bootstrap server certificate is expired, or is due to
10 expire within a threshold time period, for example a month, a week, a day etc., the certificate expiry checker may be configured to initiate replacement of the relevant bootstrap server certificate. Replacement of the relevant bootstrap server certificate may be achieved during the bootstrapping process or EST process, as will be familiar to the skilled person. In alternative arrangements, the certificate may be replaced
15 using a firmware update mechanism or other tool.

In exemplary methods, the certificate expiry checker may be configured to initiate replacement of the relevant bootstrap server certificate with a certificate with an expiry date that is later than the expiry date of the next bootstrap server in the hierarchy or
20 sequence. Taking the example outlined in Figure 5, in which the hierarchy or sequence defines that the reconfiguration order is: the final bootstrap server 106b, the intermediate bootstrap server 106c and the first bootstrap server 106a, if the final bootstrap server certificate has expired and the intermediate bootstrap server certificate expiry date is in one years' time, then the certificate expiry checker may be
25 configured to initiate replacement of the final bootstrap server 106b with a certificate that expires later than one years' time. This ensures that the full reconfiguration hierarchy or sequence can be initiated if necessary.

The certificate expiry checker may be configured to check the certificate expiry date
30 resources at pre-defined intervals, for example, each day, week, month, etc.

It will be appreciated by the person of skill in the art that various modifications may be made to the above described embodiments without departing from the scope of the invention. The word "exemplary" is used herein to mean "an example". Any

embodiment described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other embodiments.

5 A computer program may be configured to provide any of the above described methods. The computer program may be provided on a computer readable medium. The computer program may be a computer program product. The product may comprise a non-transitory computer usable storage medium. The computer program product may have computer-readable program code embodied in the medium configured to perform the method. The computer program product may be configured to cause at least one processor to perform some or all of the method.

15 Various methods and apparatus are described herein with reference to block diagrams or flowchart illustrations of computer-implemented methods, apparatus (systems and/or devices) and/or computer program products. It is understood that a block of the block diagrams and/or flowchart illustrations, and combinations of blocks in the block diagrams and/or flowchart illustrations, can be implemented by computer program instructions that are performed by one or more computer circuits. These computer program instructions may be provided to a processor circuit of a general purpose computer circuit, special purpose computer circuit, and/or other programmable data processing circuit to produce a machine, such that the instructions, which execute via the processor of the computer and/or other programmable data processing apparatus, transform and control transistors, values stored in memory locations, and other hardware components within such circuitry to implement the functions/acts specified in the block diagrams and/or flowchart block or blocks, and thereby create means (functionality) and/or structure for implementing the functions/acts specified in the block diagrams and/or flowchart block(s).

30 Computer program instructions may also be stored in a computer-readable medium that can direct a computer or other programmable data processing apparatus to function in a particular manner, such that the instructions stored in the computer-readable medium produce an article of manufacture including instructions which implement the functions/acts specified in the block diagrams and/or flowchart block or blocks.

A tangible, non-transitory computer-readable medium may include an electronic, magnetic, optical, electromagnetic, or semiconductor data storage system, apparatus, or device. More specific examples of the computer-readable medium would include the following: a portable computer diskette, a random access memory (RAM) circuit, a
5 read-only memory (ROM) circuit, an erasable programmable read-only memory (EPROM or Flash memory) circuit, a portable compact disc read-only memory (CD-ROM), and a portable digital video disc read-only memory (DVD/Blu-ray).

The computer program instructions may also be loaded onto a computer and/or other
10 programmable data processing apparatus to cause a series of operational steps to be performed on the computer and/or other programmable apparatus to produce a computer-implemented process such that the instructions which execute on the computer or other programmable apparatus provide steps for implementing the functions/acts specified in the block diagrams and/or flowchart block or blocks.

15 Accordingly, the invention may be embodied in hardware and/or in software (including firmware, resident software, micro-code, etc.) that runs on a processor, which may collectively be referred to as "circuitry," "a module" or variants thereof.

20 It should also be noted that in some alternate implementations, the functions/acts noted in the blocks may occur out of the order noted in the flowcharts. For example, two blocks shown in succession may in fact be executed substantially concurrently or the blocks may sometimes be executed in the reverse order, depending upon the functionality/acts involved. Moreover, the functionality of a given block of the flowcharts
25 and/or block diagrams may be separated into multiple blocks and/or the functionality of two or more blocks of the flowcharts and/or block diagrams may be at least partially integrated. Finally, other blocks may be added/inserted between the blocks that are illustrated.

30

CLAIMS:

1. A method of managing bootstrapping of a client of a utility metering device in a Lightweight Machine to Machine (LwM2M) system, wherein bootstrapping comprises
5 configuring the client for connection and registration with one or more device management servers, the method comprising:
- configuring the client with a security object comprising a first security object instance associated with a first bootstrap server, the first security object instance comprising data for use by the client in establishing communication with the first
10 bootstrap server;
- configuring the client, by the first bootstrap server, with a final bootstrap security object instance associated with a final bootstrap server and comprising data for use by the client in establishing communication with the final bootstrap server, the final bootstrap server for configuring the client for connection and registration with one or
15 more device management servers;
- establishing, by the client and with the final bootstrap server, communication using the final bootstrap security object instance; and
- configuring, by the final bootstrap server, the client for connection and registration with the one or more device management servers,
- 20 wherein if the client is unable to connect and/or register with the one or more device management servers, the method further comprises initiating communication between the client and the final bootstrap server for reconfiguration of the client by the final bootstrap server for a subsequent connection and/or registration attempt with the one or more device management servers.
- 25
2. A method according to claim 1, wherein configuring, by the final bootstrap server, the client for connection and registration with the one or more device management servers comprises: configuring the client with a device management security object instance associated with the one or more device management servers
30 and comprising data for use by the client in communicating with the one or more device management servers.

3. A method according to claim 2, wherein the reconfiguration of the client by the final bootstrap server comprises configuring the client with an updated device management security object instance.

5 4. A method according to any preceding claim, wherein if the client is unable to connect and/or register with the one or more device management servers following the reconfiguration by the final bootstrap server, the method further comprises initiating communication between the client and the first bootstrap server for reconfiguration of the client by the first bootstrap server.

10 5. A method according to any preceding claim, wherein the first bootstrap server configures clients of a plurality of utility metering devices, and wherein the final bootstrap server configures clients of a subset of the plurality of utility metering devices.

15 6. A method according to claim 1, wherein the first bootstrap server configures the client with an intermediate bootstrap security object instance associated with an intermediate bootstrap server and comprising data for use by the client in communicating with the intermediate bootstrap server, and wherein the method further comprises:

20 establishing, by the client and with the intermediate bootstrap server, data communication using the intermediate bootstrap security object instance; and

25 configuring the client, by the intermediate bootstrap server, with the final security object instance associated with the final bootstrap server such that the client can establish data communication with the final bootstrap server for connection and registration with the one or more device management servers.

30 7. A method according to claim 6, wherein if the client is unable to connect and/or register with the one or more device management servers following the reconfiguration of the client by the final bootstrap server, the method further comprises determining a sequence in which communication should be initiated with the final bootstrap server, the intermediate bootstrap server and the first bootstrap server for reconfiguration of the client by each of the final bootstrap server, the intermediate bootstrap server and the first bootstrap server.

35

8. A method according to claim 7, wherein the client initiates communication for reconfiguration with one of the final bootstrap server, the intermediate bootstrap server and the first bootstrap server based on the determined sequence, and wherein communication with a next bootstrap server in the determined sequence is only
5 initiated if, after reconfiguration by the one of the final bootstrap server, the intermediate bootstrap server and the first bootstrap server, registration with the one or more device management servers fails.
9. A method according to claim 6, wherein if the client is unable to connect and/or
10 register with the one or more device management servers following the reconfiguration of the client by the final bootstrap server, the method further comprises:
initiating communication between the client and the intermediate bootstrap server for reconfiguration of the client by the intermediate bootstrap server; and
wherein if the client is unable to connect and/or register with the one or more
15 device management servers following the reconfiguration of the client by the intermediate bootstrap server, the method comprises initiating communication between the client and the first bootstrap server for reconfiguration of the client by the first bootstrap server.
10. A method according to any of claims 6 to 9, wherein the first bootstrap server
20 configures clients of a plurality of utility metering devices, the intermediate bootstrap server configures clients of a subset of the plurality of utility metering devices, and wherein the final bootstrap server configures clients of at least one of the subset of the plurality of metering devices.
11. A method according to any preceding claim, wherein if the final bootstrap
25 security object instance comprises data indicating that an enrolment over secure transport, EST, certificate is needed for registration with the one or more device management servers, the method further comprises:
30 sending, by the client and to an EST server, an enrolment over secure transport, EST, certificate request; and
receiving, by the client and from the EST server, the EST certificate;
using the EST certificate for connection and/or registration with the one or more device management servers.

12. A method according to any preceding claim, wherein the client comprises a first bootstrap server certificate for authenticating communication between the client and the first bootstrap server, and a final bootstrap server certificate for authenticating communication between the client and the final bootstrap server, and wherein the method comprises:
- 5 determining, by the client, an expiry date of the first bootstrap server certificate and/or the final bootstrap server certificate; and
- initiating replacement of the first bootstrap server certificate and/or the final bootstrap server certificate if the respective expiry dates fall within a threshold time
- 10 period.
13. A method according to claim 12, when dependent on any one of claims 6 to 11, wherein the client further comprises an intermediate bootstrap server certificate for authenticating communication between the client and the intermediate bootstrap
- 15 server, and wherein the method further comprises:
- determining, by the client, an expiry date of the intermediate bootstrap server certificate; and
- initiating replacement of the intermediate bootstrap server certificate if the expiry date falls within the threshold time period.
- 20
14. A method according to any preceding claim, wherein the first bootstrap server is configured to configure the client on a first boot-up of the client.