

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: March 19, 2011

T. Tsou
Huawei Technologies
J. Schoenwaelder
Jacobs University Bremen
Y. Shi
Hangzhou H3C Tech. Co., Ltd.
T. Taylor, Ed.
Huawei Technologies
September 15, 2010

Problem Statement for the Configuration of Large-Scale IP Networks
draft-tsou-opsawg-network-configuration-01

Abstract

This memo discusses the steps required to bring network devices in a service provider network into service in an automated fashion. The memo identifies known solutions where they exist, but notes some gaps that require further specification.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on March 19, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

Table of Contents

1.	Introduction	3
2.	Scenarios	4
3.	A Model of the Process	5
4.	Pre-configuration	6
5.	Bootstrapping	6
5.1.	Establishment of Link Layer Connectivity	6
5.2.	Acquisition of IP Addresses and Basic Routing Information	6
5.3.	Finding the Configuration Server	7
5.4.	Establishing a Secure Channel to the Configuration Server	8
6.	Initial Configuration and Configuration Updates	10
7.	Configuration Auditing	12
8.	Missing Specifications	13
9.	Security Considerations	14
10.	IANA Considerations	14
11.	Contributors	14
12.	Informative References	14
Appendix A.	Open Issues	18
	Authors' Addresses	18

1. Introduction

New service provider networks are being deployed that entail the installation of tens of thousands of new network devices. To keep costs down, it is desirable to automate the establishment of such networks and the configuration of these network devices to the maximum extent possible. A certain amount of the information needed to operate them must be pre-configured by the vendor or service provider before the devices are physically deployed. Other information is best delivered after startup, to ensure that it is consistent with the physical deployment.

3GPP work in progress describes requirements [[TS 32 500](#)] and an architectural specification [[TS 36 300](#)] for the self-configuration of edge node entities called eNodeBs. (The expansion of eNodeB is too unwieldy to spell out.) Specifically, procedures are specified for establishing transport connections to and for exchanging configuration data with control entities called MMEs (Mobility Management Entities) and with neighbouring eNodeBs. [[TS 36 300](#)] currently assumes as a starting precondition that the eNodeB knows its own IP address and knows IP address endpoints for the target MMEs and neighbouring eNodeBs.

IETF work on automated configuration goes back to BOOTP [[RFC0951](#)], followed eight years later by DHCP [[RFC1531](#)] and successors. The years since have seen a steady growth in the number of DHCP options.

The Simple Network Management Protocol (SNMP) [[RFC3410](#)] was designed to convey management information between SNMP entities such as managers and agents. The number of SNMP MIB modules grew steadily, but SNMP has not historically seen only limited use for configuration [[RFC3535](#)]. For a period, IETF configuration efforts were focussed on the distribution of policy in the network. [[RFC3139](#)] provides a good insight into this period. More recently, NETCONF [[RFC4741](#)] was devised as an alternative to SNMP, but the development of standard NETCONF data models is just beginning.

Recent IETF work closest in spirit to the 3GPP self-organizing network effort cited above is embodied in CAPWAP [[RFC5415](#)]. Like the 3GPP work, CAPWAP focusses on the configuration of edge nodes, in a Wi-Fi rather than cellular network. The CAPWAP work goes beyond that of 3GPP by specifying the process of AC (Access Controller) discovery rather than leaving discovery out of scope. With regard to the configuration process itself, CAPWAP provides for the download of new images to the WTP (Wireless Termination Point). In contrast, [[TS 32 500](#)] assumes that this has already been completed for the eNodeB.

2. Scenarios

There are two different scenarios to consider. In the first scenario, called the Intra-domain Scenario, the new network device N is attached to the network operated by the service provider which is also operating the new device. In the second scenario, called the Inter-domain Scenario, the new device N is attached to a third party network providing connectivity to the network of the service provider operating the new device.

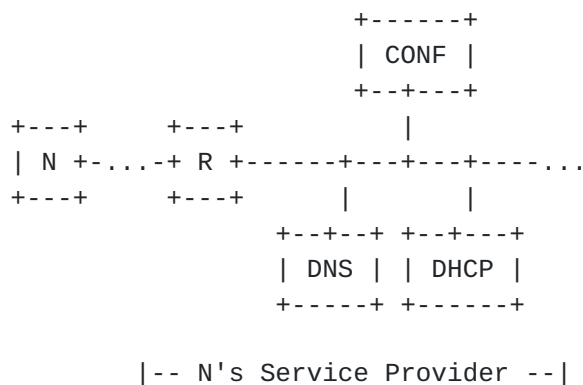


Figure 1: Intra-domain Scenario

Figure 1 depicts the Intra-domain Scenario. We assume that the new device N attaches to a link connected to router R. Furthermore, we assume that the service provider provides a Domain Name System (DNS) server, a DHCP server, and a Configuration Server (CONF). Overall, this scenario does not differ much from conventional network scenarios.

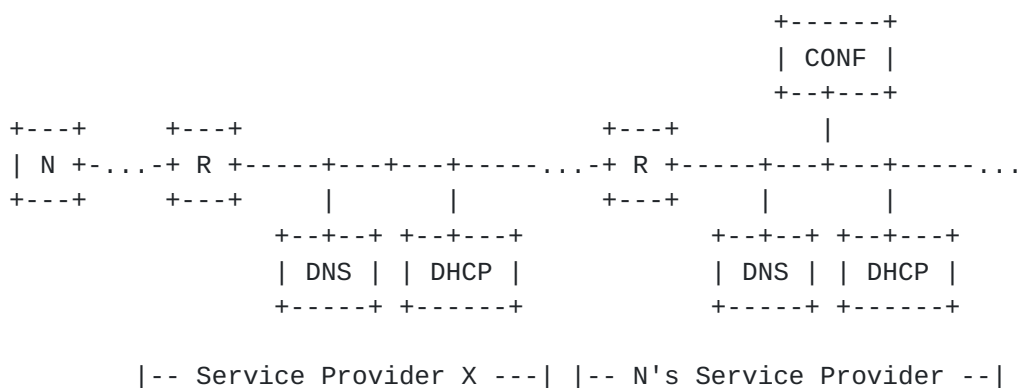


Figure 2: Inter-domain Scenario

Figure 2 depicts the Inter-domain Scenario where the new device N attaches to a router R owned by a different service provider X. The service provider X might offer its own DNS and DHCP services. We

assume that the service provider X has connectivity to the service provider planning to operate the new device.

3. A Model of the Process

We introduce a model of the configuration process in order to identify the parts that have well-known solutions. The remainder may be worth studying to see if the industry can agree on a solution.

Some basic terminology is needed for the discussion. Depending on the implementation, let us agree that "configuration data" consist of software and sets of configured parameters in some combination. Also, the system that provides the configuration data is called the "configuration server". Finally, the term "joining device" is used to denote a network device that is in the process of being incorporated into the network.

Broadly speaking, the configuration process can be broken into five phases:

Pre-configuration: configuration carried out either by the vendor or by the service provider prior to physical installation. One possible example is the pre-provisioning of certificates, as described in [[RFC5415](#)].

Bootstrapping: the portion of the process from the time that physical installation is complete until a secure connection is established between the joining device and the configuration server.

Initial configuration: downloading of the configuration data that the joining device needs to carry out its function in the network.

Auditing of installed configuration: tracking image versions and configuration parameters for each network device and verifying that the installed configuration data matches the physical installation, the network plan, and the records of what data was downloaded. It is possible that an initial audit of the physical installation is done before initial configuration, so that the validity of the intended download can be verified.

Configuration update: transferring configuration data to a fully configured and operating device from time to time as the need arises.

4. Pre-configuration

This memo identifies a specific requirement for pre-configuration of an invariant device identity and authentication-related material in the form of pre-shared secrets or certificates. There is, as one alternative, a requirement for pre-configuration of information that permits the joining device to discover the address of the configuration server.

5. Bootstrapping

[I-D.oflynn-core-bootstrapping] deals with the process of bootstrapping, with particular emphasis on the requirements for highly resource-constrained devices. The document makes a distinction between a data channel, which is used during network operation, and a control channel, which is used during bootstrapping. While both channels can be the same physical channel, they can also be different (e.g., a wireless access point using an infrared control channel to receive bootstrapping information). The draft proposes to define a generic secure bootstrapping protocol for resource constrained devices that can be executed in several bootstrapping rounds and can be adapted to the specific contexts in terms of the resources available within individual devices and for the network as a whole.

For network devices in service provider networks, bootstrapping consists of several stages:

1. establishment of link layer connectivity with neighbouring nodes;
2. acquisition of IP addresses and basic routing information;
3. discovery of the configuration server;
4. establishment of a secure channel to the configuration server.

5.1. Establishment of Link Layer Connectivity

The protocol aspects of this phase are out of scope, since it involves non-IETF protocols only. While some link-layer technologies may provide authentication and access control, this cannot be assumed to be available in the general case.

5.2. Acquisition of IP Addresses and Basic Routing Information

For IPv4, DHCPv4 [[RFC2131](#)] is widely deployed and the usual way to obtain an IPv4 address, the IPv4 address of a link-local router and

the IPv4 address of a DNS server. For IPv6, a choice has to be made between stateful DHCPv6 [[RFC3315](#)] versus stateless DHCPv6 [[RFC3736](#)] combined with stateless address autoconfiguration [[RFC4862](#)]. In the latter case, DHCPv6 is needed to configure parameters such as DNS server addresses. An experimental routing advertisement option to configure the IPv6 address of a DNS server as part of the stateless address autoconfiguration is defined in [[RFC5006](#)] and may become a standards-track specification.

Some security protection is provided in this stage by using DHCP authentication [[RFC3118](#)]. However, security of the configuration process as a whole has to be assured by other means. This is discussed further below.

Currently the lack of a stable identifier for use in DHCPv6 messaging is an impediment to authentication of the joining device. [[I-D.dhc-duid-uuid](#)] discusses the problems with the current DHCPv6 identifiers (DUIDs) and proposes a new form that could be a more stable alternative.

[5.3.](#) Finding the Configuration Server

Four alternatives are available for finding the configuration server:

- o pre-configuration;
- o DHCP configuration;
- o Service Location Protocol [[RFC2608](#)]; or
- o DNS service discovery using SRV records [[RFC2782](#)].

Pre-configuration of an IP address is brittle and not recommended. Pre-configuration of a Uniform Resource Identifier (URI) or fully qualified domain name (FQDN) is a better approach. One variant that has been suggested is to burn the URI of a vendor server into the device's firmware along with a device identifier, and have that server redirect to the URI of the service provider's configuration server based on the device identity. Such an approach requires that a device vendor offers such a service for the lifetime of their devices and that service providers are able to update the URI of the service provider's configuration server. This requires a trust relationship between the vendor and the service provider and agreement on a protocol to update the redirect information on the vendor's server.

DHCP configuration can use the usual DHCP options and is technically straightforward since DHCP is widely used by end user devices to

obtain basic configuration information. There is, however, no standardized DHCP option to communicate the address of a configuration server.

The Service Location Protocol (SLP) has seen some usage to locate services such as printers or file system shares. Usage of SLP to locate configuration servers requires to define a new service template [[RFC2609](#)].

The use of DNS SRV records requires the joining device to obtain the correct domain suffix first, presumably from DHCP or via Routing Advertisements in the case of IPv6 or preconfiguration. A service type for the desired configuration protocol would have to be defined in the DNS for the purpose. See [Section 3.3 of \[RFC5415\]](#) for a discussion of the corresponding discovery process for CAPWAP.

The Inter-domain Scenario requires that the DHCP server or the SLP server of service provider X's network is able to provide the correct information to the joining devices. To accomplish this, the discovery servers need to be able to match a device identification against a list of possible configuration servers. Furthermore, there needs to be a mechanism for the service provider operating the joining device to provision the configuration server's address, e.g. by using an extension of the Extensible Provisioning Protocol (EPP) [[RFC5730](#)]. However, if the joining device has preconfigured information about the name of the service provider's network, DNS SRV records may be queried after obtaining IP connectivity, avoiding the need to provision information in service provider X's network.

[5.4.](#) Establishing a Secure Channel to the Configuration Server

It is essential that the configuration server and the joining device authenticate themselves to each other, since the steps leading up to this point in the process may not be fully secure. This raises two issues: how the joining device identifies itself, and how authentication takes place.

It seems best if the device has an invariant identity built in and accessible to whatever operating system is running on it. If [[I-D.dhc-duid-uuid](#)], mentioned above, becomes a standard, the UUID on which that proposal is based would be the required invariant identity. The vendor should make that identity available in a form that can be read and transferred into a database accessible to the configuration server along with the associated configuration data in advance of the bootstrapping stage (e.g., in bar-coded format on the device packaging).

This leaves the mutual authentication process itself. This has two

aspects: the security protocol used to perform authentication, and initial keying methodology. The security protocol is tied together with the choice of configuration data transport, but the basic choices are:

- o IP Security (IPsec) [[RFC4301](#)];
- o Transport Layer Security (TLS) [[RFC5246](#)];
- o Datagram Transport Layer Security (DTLS) [[RFC4347](#)];
- o Secure Shell (SSH) [[RFC4251](#)], [[RFC4252](#)], [[RFC4253](#)], and [[RFC4254](#)]; and
- o SNMPv3's User-based Security Model (USM) [[RFC3414](#)].

For initial keying methodology, the two basic choices are between pre-shared secrets and certificates. All of the security protocols listed above except USM support both methods. USM supports pre-shared secrets only.

The usual concern with pre-shared secrets is scalability. In the bootstrapping case, the scale of operation required is linear with the number of devices to be configured, so it would definitely be a feasible approach if connection to the configuration system were the only consideration. The most likely procedure would be for the secret to be configured in the device during preconfiguration and also captured in a database along with the device identity, for use by the configuration server.

The problem with the use of pre-shared secrets is that the device needs to authenticate itself at an earlier stage, while it is establishing communications with its neighbours and acquiring IP addresses. It seems undesirable to use the same secret for that purpose as for the connection to the configuration server, on the basic principle of limiting the potential damage from disclosure of a particular key.

This need for additional pre-shared secrets argues for consideration of certificates as an alternative. One issue for certificates is where the trust anchor resides. It seems logical that it should reside with the service provider rather than the vendor, to make it easy to install equipment from multiple vendors. On that basis, preconfiguration requires service provider input.

CAPWAP ([Section 2.4.4.3 of \[RFC5415\]](#)) makes use of the Extended Key Usage (EKU) certificate extension [[RFC5280](#)] to distinguish certificates identifying the Access Controllers (i.e., the

configuration servers in the CAPWAP case) from the Wireless Transfer Points (the configured devices in the CAPWAP case). Thought should be given to whether such distinctions are required in the general case of network device configuration.

CAPWAP ([Section 12.8 of \[RFC5415\]](#)) also discusses the use of the Common Name rather than SubjectAltName field of the certificate to carry device identity, due to lack of specifications allowing the use of SubjectAltName to carry MAC addresses. This issue needs to be investigated further if another form of device unique identity is used, as discussed above.

6. Initial Configuration and Configuration Updates

As mentioned at the beginning, the configuration data being downloaded may be a combination of software and configuration parameters. Some of the data will be vendor-specific, not subject to standardization. It appears that there is a continuing debate on whether the configuration data should be pushed to the joining device or whether the device should pull the configuration data down. In the latter case, the device needs to know about the existence of the data and the path to reach it before it can act. One way to acquire this information is through DHCP. DHCPv4 has provided the necessary options from its beginnings, inheriting them from BOOTP. They are currently being added to DHCPv6; see [\[I-D.dhcpv6-opt-netboot\]](#).

Protocols that can transport configuration data can be classified as follows: The first class consists of generic file transfer protocols that can carry configuration data serialized into configuration files. The second class consists of protocols that manipulate structured configuration data directly. The structure of the configuration data is defined by some data model.

In the first class, we find the following file transfer protocols:

- o The File Transfer Protocol (FTP) [\[RFC0959\]](#) can be used to move files containing configuration data. It can be secured by running FTP over TLS [\[RFC4217\]](#).
- o The Trivial File Transfer Protocol (TFTP) [\[RFC1350\]](#) has been used extensively to load boot images over the network. However, it does not provide security and the only option is to rely on IP layer security (IPsec).
- o The Hypertext Transfer Protocol (HTTP) [\[RFC2616\]](#) can be used to transfer documents containing configuration data. It is commonly secured by running HTTP over TLS [\[RFC2817\]](#) [\[RFC2818\]](#).

- o The SSH File Transfer Protocol (SFTP) [[I-D.SFTP](#)] provides roughly the same services as FTP but runs over SSH and thus utilizes the security services provided by SSH.
- o UNIX utilities to transfer files such as RCP and SCP provide limited flexibility and they differ in their degree of integration with SSH.
- o The Control And Provisioning of Wireless Access Points (CAPWAP) protocol [[RFC5415](#)] can be used to control the download of images. CAPWAP can be secured by running CAPWAP over DTLS.

In the second class, we find the following configuration protocols:

- o Version 3 of the Simple Network Management Protocol (SNMPv3) [[RFC3411](#)]-[[RFC3418](#)] can be used to manipulate MIB objects and to carry event notifications. It has its own security protocol (USM) but can also run over SSH [[RFC5592](#)], TLS, or DTLS [[RFC5953](#)].
- o The Common Open Policy Service for Policy Provisioning protocol (COPS-PR) [[RFC3084](#)] was designed to provision structured policy information from a Policy Decision Point (PDP) to a Policy Enforcement Point (PEP). The COPS protocol [[RFC2748](#)] provides an integrity object that can achieve authentication, message integrity, and replay prevention. Optionally, COPS and COPS-PR can run over TLS.
- o The NETCONF protocol [[RFC4741](#)] provides mechanisms to install, manipulate, and delete the configuration of network devices. A protocol extension provides an asynchronous even notification delivery mechanism [[RFC5277](#)]. NETCONF by default runs over SSH but can also run over transports secured by TLS.
- o The Control And Provisioning of Wireless Access Points protocol (CAPWAP) [[RFC5415](#)] supports the discovery of so called Access Controller (AC) by Wireless Termination Points (WTPs) and the configuration of WTPs by an AC. While CAPWAP can be extended to configure other devices, its main focus are WTPs. The CAPWAP protocol is protected by using DTLS after the discovery phase.

Table 1 lists the protocols plus their security options. Note that all protocols can be secured at the IP layer by using IPsec and hence this is not mentioned explicitly in Table 1.

Transport	Security	Data Transfer Model
FTP	TLS	Push or pull of (configuration) files
TFTP		Push or pull of (configuration) files
HTTP	TLS	Push or pull of (configuration) files
SFTP	SSH	Push or pull of (configuration) files
RCP		Push or pull of (configuration) files
SCP	SSH	Push or pull of (configuration) files
CAPWAP	DTLS	AC pushes configuration parameters, WTP pulls software
SNMPv3	USM [SSH, TLS, DTLS]	Push of structured configuration parameters, event notifications
COPS-PR	TLS	Push of structured policy information
NETCONF	SSH [TLS]	Push of structured configuration data, event notifications

Table 1: Protocols for transporting configuration data

SNMPv3, NETCONF, and COPS-PR carry structured data specified in pre-defined data models. SNMPv3 and COPS-PR have size limitations on the data objects and thus make the transport of larger software images difficult. NETCONF does not suffer from hard size restrictions and can in principle carry software images inline. However, there is currently no work in progress to standardize the transfer of software images over NETCONF. An advantage of NETCONF over SNMPv3 and CAPWAP is the support of concurrent updates through locking mechanisms and the support of network wide configuration change transactions through the confirmed commit capability. CAPWAP combines the functions of configuration parameter transport and software download. The parameter transport aspect lacks the generality offered by SNMP, NETCONF, and COPS-PR, since the parameters are specified within the protocol specification itself. The remaining transports are independent of the nature of the information being transferred.

7. Configuration Auditing

To complete the process, it must be possible to audit the configuration status of the device in some detail. This is likely to begin even before all the configuration data has been downloaded. For instance, configuration management may wish to collect basic information such as the MAC addresses of the device's interfaces, the link-local addresses assigned to them, and similar information for the neighbours of the joining device.

SNMP and SNMP MIB modules are obviously one way to collect this

information. NETCONF [[RFC4741](#)] is an alternative, but the necessary data models have to be defined. YANG modules for NETCONF [[I-D.YANG](#)] can be prepared relatively quickly from existing SNMP MIB modules by translating the SNMP modules into YANG modules. Work to standardize such translations is currently being chartered in the NETMOD working group.

Another important auditing activity is the analysis of system events. The SYSLOG protocol [[RFC5424](#)] is widely used for this purpose but SNMPv3 and NETCONF can ship event notifications as well. Translations of SNMP notifications into structured SYSLOG messages and vice versa do exist [[RFC5675](#)] [[RFC5676](#)]. NETCONF can carry SYSLOG content as well [[RFC5277](#)].

8. Missing Specifications

This document discussed the automated configuration of devices in service provider networks. Several gaps were identified requiring further specification:

- G1: Definition of stable unique device identifiers such as the work described in [[I-D.dhc-duid-uuid](#)].
- G2: Definition of a DHCP option to provide the IPv4/IPv6 address of a configuration server. Such an option allows a joining device to pickup the configuration server's address as part of the DHCP exchange. This is particularly interesting for Intra-domain Scenarios.
- G3: Definition of DNS SRV records for locating configuration servers. Such an option allows a joining device to lookup the configuration server's in the DNS; this is particularly useful in an Inter-domain Scenario.
- G4: Definition of a SLP template for discovering configuration servers. Such a template is useful only in environments where SLP is used also for other purposes.
- G5: Definition of NETCONF data models to support the download / update of software images through NETCONF.
- G6: Definition of NETCONF data models for collecting basic system information and integrity information (e.g., checksums of software images) and for sending configuration management related notifications.

9. Security Considerations

The security of a configuration management solution is of crucial importance. [Section 6](#) discusses the security options of several protocols that might be used. The relevant protocol definitions should be consulted to learn more about the specific security aspects of the various protocols.

It should be noted that some steps in the described process, in particular the bootstrapping phase, may not be secure and it is thus important to verify the identity of the device and the identity of the configuration server when a secure connection to a configuration server is established. Usage of IPsec, which is focusses on securing the IP layer, may not be sufficient for this.

During the choice of protocols, the available security mechanisms and the required key management infrastructures may play a major role in the selection of protocols. Easy integration into existing Authentication, Authorization and Accounting (AAA) infrastructures can significantly reduce the operational costs associated with the security management of the configuration system.

10. IANA Considerations

This memo includes no request to IANA.

11. Contributors

Thanks to Cathy Zhou and Mehmet Ersue for help in preparing this memo.

12. Informative References

[I-D.SFTP]

Galbraith, J. and O. Saarenmaa, "SSH File Transfer Protocol (Work in progress)", July 2006.

<http://tools.ietf.org/id/draft-ietf-secsh-filexfer-13.txt>

[I-D.YANG]

Bjorklund, M., "YANG - A data modeling language for the Network Configuration Protocol (NETCONF) (Work in progress)", June 2010.

[I-D.dhc-duid-uuid]

Narten, T. and J. Johnson, "Definition of the UUID-based DHCPv6 Unique Identifier (DUID-UUID) (Work in progress)", May 2010.

[I-D.dhcpv6-opt-netboot]

Huth, T., Freimann, J., Zimmer, V., and D. Thaler, "DHCPv6 option for network boot (Work in progress)", June 2010.

[I-D.oflynn-core-bootstrapping]

O'Flynn, C. and B. Sarikawa, "Initial Configuration of Resource-Constrained Devices (Work in progress)", July 2010.

[RFC0951] Croft, B. and J. Gilmore, "Bootstrap Protocol", [RFC 951](#), September 1985.

[RFC0959] Postel, J. and J. Reynolds, "File Transfer Protocol", STD 9, [RFC 959](#), October 1985.

[RFC1350] Sollins, K., "The TFTP Protocol (Revision 2)", STD 33, [RFC 1350](#), July 1992.

[RFC1531] Droms, R., "Dynamic Host Configuration Protocol", [RFC 1531](#), October 1993.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", [RFC 2131](#), March 1997.

[RFC2608] Guttman, E., Perkins, C., Veizades, J., and M. Day, "Service Location Protocol, Version 2", [RFC 2608](#), June 1999.

[RFC2609] Guttman, E., Perkins, C., and J. Kempf, "Service Templates and Service: Schemes", [RFC 2609](#), June 1999.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.

[RFC2748] Durham, D., Boyle, J., Cohen, R., Herzog, S., Rajan, R., and A. Sastry, "The COPS (Common Open Policy Service) Protocol", [RFC 2748](#), January 2000.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", [RFC 2782](#), February 2000.

[RFC2817] Khare, R. and S. Lawrence, "Upgrading to TLS Within

HTTP/1.1", [RFC 2817](#), May 2000.

- [RFC2818] Rescorla, E., "HTTP Over TLS", [RFC 2818](#), May 2000.
- [RFC3084] Chan, K., Seligson, J., Durham, D., Gai, S., McCloghrie, K., Herzog, S., Reichmeyer, F., Yavatkar, R., and A. Smith, "COPS Usage for Policy Provisioning (COPS-PR)", [RFC 3084](#), March 2001.
- [RFC3118] Droms, R. and W. Arbaugh, "Authentication for DHCP Messages", [RFC 3118](#), June 2001.
- [RFC3139] Sanchez, L., McCloghrie, K., and J. Saperia, "Requirements for Configuration Management of IP-based Networks", [RFC 3139](#), June 2001.
- [RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", [RFC 3315](#), July 2003.
- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", [RFC 3410](#), December 2002.
- [RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, [RFC 3411](#), December 2002.
- [RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, [RFC 3414](#), December 2002.
- [RFC3418] Presuhn, R., "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)", STD 62, [RFC 3418](#), December 2002.
- [RFC3535] Schoenwaelder, J., "Overview of the 2002 IAB Network Management Workshop", [RFC 3535](#), May 2003.
- [RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", [RFC 3736](#), April 2004.
- [RFC4217] Ford-Hutchinson, P., "Securing FTP with TLS", [RFC 4217](#), October 2005.
- [RFC4251] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Protocol Architecture", [RFC 4251](#), January 2006.

- [RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", [RFC 4252](#), January 2006.
- [RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", [RFC 4253](#), January 2006.
- [RFC4254] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Connection Protocol", [RFC 4254](#), January 2006.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", [RFC 4301](#), December 2005.
- [RFC4347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security", [RFC 4347](#), April 2006.
- [RFC4741] Enns, R., "NETCONF Configuration Protocol", [RFC 4741](#), December 2006.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", [RFC 4862](#), September 2007.
- [RFC5006] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Option for DNS Configuration", [RFC 5006](#), September 2007.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), August 2008.
- [RFC5277] Chisholm, S. and H. Trevino, "NETCONF Event Notifications", [RFC 5277](#), July 2008.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", [RFC 5280](#), May 2008.
- [RFC5415] Calhoun, P., Montemurro, M., and D. Stanley, "Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification", [RFC 5415](#), March 2009.
- [RFC5424] Gerhards, R., "The Syslog Protocol", [RFC 5424](#), March 2009.
- [RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure Shell Transport Model for the Simple Network Management Protocol (SNMP)", [RFC 5592](#), June 2009.
- [RFC5675] Marinov, V. and J. Schoenwaelder, "Mapping Simple Network Management Protocol (SNMP) Notifications to SYSLOG

Messages", [RFC 5675](#), October 2009.

- [RFC5676] Schoenwaelder, J., Clemm, A., and A. Karmakar,
"Definitions of Managed Objects for Mapping SYSLOG
Messages to Simple Network Management Protocol (SNMP)
Notifications", [RFC 5676](#), October 2009.
- [RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)",
STD 69, [RFC 5730](#), August 2009.
- [RFC5953] Hardaker, W., "Transport Layer Security (TLS) Transport
Model for the Simple Network Management Protocol (SNMP)",
[RFC 5953](#), August 2010.
- [TS_32_500]
3rd Generation Partnership Project, "3rd Generation
Partnership Project; Technical Specification Group
Services and System Aspects; Telecommunication Management;
Self-Organizing Networks (SON); Concepts and requirements
(Release 9)", 3GPP TS 32.500, 2010.
- [TS_36_300]
3rd Generation Partnership Project, "3rd Generation
Partnership Project; Technical Specification Group Radio
Access Network; Evolved Universal Terrestrial Radio Access
(E-UTRA) and Evolved Universal Terrestrial Radio Access
Network (E-UTRAN); Overall description; Stage 2 (Release
9)", 3GPP TS 36.300, 2010.

[Appendix A.](#) Open Issues

The document should discuss the usage of VPNs in the Inter-domain scenario.

Authors' Addresses

Tina Tsou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: tena@huawei.com

Juergen Schoenwaelder
Jacobs University Bremen
Campus Ring 1
Bremen 28759
Germany

Email: j.schoenwaelder@jacobs-university.de

Yang Shi
Hangzhou H3C Tech. Co., Ltd.
Beijing R&D Center of H3C, Digital Technology Plaza,
NO.9 Shangdi 9th Street, Haidian District,
Beijing
China(100085)

Phone: +86 010 82775276

Email: young@h3c.com

Tom Taylor (editor)
Huawei Technologies
1852 Lorraine Ave.
Ottawa K1H 6Z8
Canada

Email: tom111.taylor@bell.net

