

ANIMA
Internet-Draft
Intended status: Standards Track
Expires: January 3, 2019

T. Eckert, Ed.
Huawei
Jul 2, 2018

Autoconfiguration of NOC services in ACP networks via GRASP
draft-eckert-anima-noc-autoconfig-00

Abstract

This document defines standards for the autoconfiguration of crucial NOC services on ACP nodes via GRASP. It enables secure remote access to zero-touch bootstrapped ANI devices via SSH/Netconf with Radius/Diameter authentication and authorization and provides lifecycle autoconfiguration for other crucial services such as syslog, NTP (clock synchronization) and DNS for operational purposes.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on January 3, 2019.

Copyright Notice

Copyright (c) 2018 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 (<https://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 Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Overview	2
1.2. ACP nodes supporting NOC autoconfiguration	3
1.3. Use of ACP GRASP for autoconfiguration	3
1.4. GRASP parameters	4
2. Services	7
2.1. Syslog	7
2.2. NTP	10
2.3. DNS for operations	12
2.4. Radius	13
2.5. Diameter	14
2.6. SSH server	14
3. Security Considerations	14
4. Change log [RFC Editor: Please remove]	15
5. References	15
5.1. Normative References	15
5.2. Informative References	16
Author's Address	16

1. Introduction

1.1. Overview

This document defines standards for the autoconfiguration of NOC services on ACP nodes via GRASP.

One key purpose of this autoconfiguration is the seamless step from zero-touch bootstrap in ANI devices over to a securely remotely configurable ACP node. For this purpose, this document defines the autoconfiguration of the SSH/Netconf server on the ACP node with autoconfigured authentication across the ACP with Radius or Diameter.

Both for this initial bringup as well as for ongoing operations of ACP nodes, this document describes autoconfiguration of syslog services as well as autoconfiguration of DNS for operational purposes. Syslog is also specified to make tracking of device configuration state easier and allow to fully automate the bringup of the ACP node after zero-touch bootstrap.

1.2. ACP nodes supporting NOC autoconfiguration

This document introduces the term ACPna nodes to indicate nodes supporting ACP that also support the requirements described in this document: ACP (n)oc (a)utoconfigurable.

If an ACPna node supports zero-touch bootstrap of the ACP where no configuration is possible before the ACP is enabled, then the NOC autoconfiguration features described in this document SHOULD be enabled by default on such an ACPna node after this zero-touch bootstrap, because the autoconfiguration of these NOC services can be the only method for the ACPna node to become operationally accessible from the NOC so that it can further be configured. ANI nodes are nodes supporting ACP and BRSKI ([I-D.ietf-anima-bootstrapping-keyinfra]). BRSKI bootstrap is an instance of such a zero-touch bootstrap requiring auto-enablement of NOC autoconfiguration after zero-touch bootstrap.

If an ACPna node was not zero-touch bootstrapped, then NOC autoconfiguration SHOULD be enabled whenever ACP is enabled but may be separately configurable.

1.3. Use of ACP GRASP for autoconfiguration

Autoconfiguration of ACPna services utilizes the ACP instance of GRASP, ([I-D.ietf-anima-grasp] as defined in [I-D.ietf-anima-autonomic-control-plane]. It leverages and extends the GRASP objective definitions of [I-D.eckert-anima-grasp-dnssd]. Those objective elements allow to create DNS-SD compatible service announcements with flexible priority/weight and distance based selection across multiple service instances and per-service parameters.

Nodes in a NOC supporting a particular service announce it via the appropriate GRASP objective into ACP GRASP. The NOC nodes therefore need to have access to the ACP, either directly because they are ACP nodes or because they use the ACP connect function (see [I-D.ietf-anima-autonomic-control-plane]). ACPna nodes receive these announcements and auto-configure the services tied to them. In most instances, the service announcement from the NOC is for a server instance that a client instance on the ACPna node connects to, for example a syslog server in the NOC. In another instance, the NOC service is that of an authentication service and the ACPna nodes will enable a server instance that leverages the authentication service in the NOC.

Note: Currently, this document does not define the option of an mDNS/DNS-SD -> ACP GRASP gateway function to enable NOC nodes without

GRASP implementations to utilize mDNS/DNS-SD to announce their services and then expect an appropriate translation function to convert these announcements into GRASP objectives. This document does define all the GRASP objectives so that that it would be possible to define such a gateway function, but some loss of functionality would exist. For once, GRASP does support network distance based service selection (e.g., select a server from the closest NOC), whereas no such mechanism exists in DNS-SD. In addition, this document believes that support of GRASP software to announce services from NOC systems is very easy to accomplish.

1.4. GRASP parameters

Unless otherwise described, all GRASP objective announcements described in this document SHOULD default to the following GRASP parameters. These parameters MAY all be configurable on the NOC nodes.

- o M_FLOOD GRASP message, periodically sent once every 60 second. Random phase vs. full minutes (so different service announcements are distributed over time in the network).
- o ttl of 210000 msec (3.5 times 60 seconds).
- o locator-option is the ACP address of the announcing node unless the announcement is done from a third-party, for example if the announcing server does not support GRASP but GRASP is run on another NOC node.
- o objective-name is 'SRV.<name>', where <name> is an [RFC6335] registered service name for the service in question.
- o objective-flags is sync-only, loop-count is 255.
- o objective-value MUST comply with the requirements of [I-D.eckert-anima-grasp-dnssd].

```

[M_FLOOD, 12340815, h'fd89b714f3db0000200000064000001', 210000,
  ["SRV.syslog", 4, 255,
    { rfcXXXX: {
      &(sender-loop-count:1) => 255,
      &(srv-element:2) => {
        &(msg-type:1) => &(describe: 0),
        &(service:2) => "syslog",
        &(instance:3) => "east-coast-primary",
        &(priority:5) => 0,
        &(weight:6) => 65535,
        &(kvpairs:7) => { "replicate" => 2 },
        &(range:8) => 2,
      }
    }
  ],
  [O_IPv6_LOCATOR,
    h'fd89b714f3db0000200000064000001', TLS12, 514]
]

```

Figure 1: SRV.syslog example

The above example shows the default values for a "syslog" service announcement using the objective-value elements defined in [I-D.eckert-anima-grasp-dnssd]. SRV.syslog is the standard objective name for the "syslog" service, as is SRV.<any> for the <any> service. The announcer of this objective also provides the syslog service as it is announcing its own address in the locator option. It provides syslog on the standard syslog TCP port 514 using TLS12.

The DNS-SD equivalent service attributes are carried in the srv-element. The msg-type indicates that this objective is a service announcement. The instance of "" indicates that this service announcement for the ACP itself, and not for e.g. the data-plane. It is shown here just for illustration purposes and can be left out in encoding because it is the default. Likewise, the service element is redundant and shown only for illustrative purposes. Priority and weight have the same semantic as in DNS-SD SRV records. In this case, the service announcement indicates the highest priority (0) and the highest weight (65535). Kvpairs includes service specific options.

Going beyond the capabilities, the range parameter indicates that the client of this service should select this announced service not only by priority/weight but primarily by the distance in terms of network hop-count between this service announcer and the client: The client is expected to select the best service announcement by priority and weight only between alternatives that are not more than two network

hops apart in distance to the client. Otherwise the client should pick the closer one.

To allow the client to know the distance to a service announcement, the sender-loop-count parameter is included in the announcement. It MUST be set by the sender to the same value (255 in this example) as the loop-count in the GRASP header. The loop-count in the header is hop-by-hop reduced. When the GRASP message arrives at the client, the difference between sender-loop-count and loop-count is the distance to the service announcer in hops.

```

;
; Following GRASP header definitions from GRASP
;
flood-message = [M_FLOOD, session-id, initiator, ttl,
                 +[objective, (locator-option / [])]]
objective = ["SRV.<rfc6335-name>", objective-flags, loop-count,
            objective-value]

objective-flags = sync-only ; as in GRASP spec
sync-only       = 4         ; M_FLOOD only requires synchronization
loop-count      = 255       ; recommended
;
; Following GRASP objective-value definitions from GRASP DNS-SD
;
objective-value = { 1*elements }
elements        = ( @rfcXXXX: { 1*relement } )
relement       /= ( &(amp)sender-loop-count:1) => 1..255 )
relement       /= ( &(amp)srv-element:2) => context-element )
context-element = {
    ?( &(amp)private:0)      => any),
    ?( &(amp)msg-type:1)     => msg-type),
    ?( &(amp)service:2)      => tstr),
    *( &(amp)instance:3)     => tstr),
    ?( &(amp)domain:4)       => tstr),
    ?( &(amp)priority:5)     => 0..65535 ),
    ?( &(amp)weight:6)       => 0..65535 ),
    *( &(amp)kvpairs:7)      => { *(tstr: any) },
    ?( &(amp)range:8)        => 0..255 ),
    *( &(amp)clocator:9)     => clocator),
}
;
TLS12 = 257

```

Figure 2: GRASP service definition

The above picture shows the complete CDDL definition of a GRASP M_FLOOD message indicating a service together with the objective-value encoding. Som of the context-element options are not used in this document (TBD - remove before going RFC).

The value 257 is defined to indicate TLS12 ([RFC5246]) to be used in the protocol field of GRASP locators to indicate that a TCP port is intended to be used with TLS version 1.2. Values 1..255 are reserved for IP protocol numbers.

2. Services

2.1. Syslog

ACPna nodes SHOULD support autoconfiguring of syslog via the SRV.syslog objective.

When an ACPna node discovers one or more SRV.syslog announcements across the ACP, it SHOULD perform syslog operations to the best available discovered server.

Local configuration of syslog on the ACPna node SHOULD have no impact on the autoconfigured syslog operations, or else, misconfiguration could cause to failure of the autoconfigured syslog operations. Instead, configured syslog operations should just operate as ships-in-the-night to the GRASP learned autoconfigured syslog operations.

Severity of syslog messages SHOULD be 5 (Notice) (see [RFC5424]), and all messages that are necessary to support normal remote operations of the node should be assigned severities higher (numerically lower) or equal to 5/Notice.

Syslog service announcements SHOULD include the instance option, indicating the unique name of the service instance described by the GRASP objective. This serves diagnostics and avoids having to identify service instances by the address(es) in the locator-options. In the example Figure 1, the instance name is "east-coast-primary".

The syslog facility value is a choice of the ACPna node, the autoconfigured syslog server must be able to deal with any syslog facility code received. If an ACPna node has no pre-established standard for the facility-code, then the value of local7 (23) MAY be used.

For resilience, it may be appropriate to receive syslog messages on more than one server. A server can indicate this via the "replicate" keyword in the GRASP objective-value kvpair element. The value of the "replicate" keyword indicates the maximum number of syslog

servers that the client SHOULD autoconfigure syslog to. After selecting the best service announcement, the client looks up the value N of the "replicate" keyword of that best servers announcement and selects the best N-1 service announcements and ultimately logs to all N. ACPna nodes SHOULD support autoconfigured syslog to up to 3 servers simultaneously.

Autoconfigured syslog SHOULD support TLS1.2, TCP and UDP. Because ACP provides encryption, use of just TCP instead of TLS should be sufficient and may achieve higher performance. Use of UDP should be avoided because of the potential to loose packets and not supporting congestion control.

If a syslog server supports more than one transport option (TLS1.2, TCP, UDP), it SHOULD announce them via a single GRASP objective and list them via clocator options of the srv-element because the locator-option in the GRASP header (as shown in example Figure 1) allows only one locator-option. The order of the clocator options in the indicates the preference of the server. From this list, the client supports the first option supported also by the client and ignore the others. The context of the clocator would normally be "", indicating that the locator-option address is reachable via the ACP.

Instead of (or in addition to) using multiple clocator options, a server can also announce multiple SRV.syslog objectives, but in that case each of them would be considered to be a different service instance considered by the the client when selecting the (set of) best service instances. If a service announcement indicates via the "replicate" keyword that the client should log to three service instances, and announce three separate SRV.syslog objectives, each one with a different locator-option, then the client might select to log to all three of them - instead of - which is more likely the desired option - for the client to log to actually three different servers. Hence the use of multiple clocator options that are examined by clients only after server selection is done.

When a client uses TLS, it MUST use its ACP domain certificate for authentication. Likewise, the syslog server MUSTS use its ACP domain certificate.

Logging by default uses the ACP, in the clocator option, this is indicated via a context value of "". Servers may also indicate support for logging across the data-plane, which may provide higher performance but may fail if reachability in the data-plane does not exist, so care must be taken when announcing this option. For example, in managed MPLS/VPN networks where the ACP extends across P/PE and CE devices, the global routing table on a CE device is often not the same as that on P/PE devices, and therefore CE devices may

not be able to log to "0". In this case, the syslog server should instead announce a deployment choosen name for the context, such as "VRF0". Clients would only take such a clocator into account if there is a local configuration that maps the context name to a routing table. In this example, only P/PE nodes would have this configuration, therefore allowing the CE nodes to ignore this clocator; And if this clocator was the only locator-option in the GRASP objective, then the whole objective MUST be ignored by the client when selecting the best possible service instance. Note that for contexts other than the ACP (""), both IPv4 and IPv6 are possible, depending on what version(s) of IP are deployed in the data-plane.

Failure to connect to a choosen service instance SHOULD be taken into accounts by clients when selecting service instances to log to. For UDP locator-options, ICMP/ICMPv6 error indications are such connection failures. For TCP/TLS connections, connection failure includes TCP and TLS failures as well as keepalive failures. When failures occur, clients should attempt to re-connect with exponential timeouts, starting with 5 seconds and staying at 320 seconds or until the GRASP service announcement expires and is not refreshed.

When connecting to a server fails, the ACPna client SHOULD connect to the next best available server in the meantime. ACPna client SHOULD support connecting to up to four service instances if any connections fail. If for example the client is logging to two service instances because 2 is indicated in the "replicate" option of the service announcements, and one fails, the client will attempt to re-connect to it while in parallel establishing syslog connection to a third-best service-instance.

When establishing connection to a new syslog service instance, ACPna clients SHOULD log with severity 5 an indication of this event, indicating its own ACP address, the ACP address and if existing instance name of the new syslog service instance and the reason. Like any other autoconfigured syslog message, this would go to all syslog connections and therefore show up on the redundant syslog servers, allowing to recognize failure of connectivity to another syslog server - and tracing of client logs across syslog servers if the client changes them.

Examples:

```
ACP: fd89:b714:f3db::0200:0000:6400:0042 start logging to:
fd89:b714:f3db::0200:0000:6400:0001/east-coast-primary,TLS reason:
starting up
```

ACP: fd89:b714:f3db::0200:0000:6400:0042 start logging to:
fd89:b714:f3db::0200:0000:6400:0001/east-coast-primary,TLS reason:
new better service instance

ACP: fd89:b714:f3db::0200:0000:6400:0042 stop logging to:
fd89:b714:f3db::0200:0000:6400:0001/east-coast-secondary,TLS reason:
connection failure

When failures to deliver syslog messages to ANY syslog servers happen, clients SHOULD track the this and indicate loss of messages via the next working syslog connection. Note that due to the possibility of ICMP/ICMPv6 errors, only the successful delivery of messages via TLS or TCP should be tracked. TBD: need to check if this can reasonably be recommended, pending on availability of e.g. TAPS API spec to know whether a TCP write was sent and acknowledged by the receiver (given how there are no reply messages in syslog).

2.2. NTP

Time synchronization is one of the most fundamental functionality for network devices for a variety of functions to work and also for diagnostics to be comparable across the network. If problems propagate fast across the network, the client generated timestamp of events in syslog messages (or other diagnostics function) allows to trace event propagation and deduce causality. This may require network clock synchronization in the order of milliseconds, something which is easily achievable in today's network devices via NTP.

ACPna nodes SHOULD support autoconfiguration of clock synchronization through NTP ([RFC5905]) with the following autoconfiguration semantics.

The GRASP objective for NTP is SRV.ntp. This does not distinguish between NTPv4 and NTPv3 because NTPv4 is fully backward compatible with NTPv3, so server and client will negotiate between these two versions.

The kvpair key "stratum" has a numeric value and indicates the stratum or level of a server in a synchronization tree. The value of 1 indicates the root of the distribution tree. Servers that synchronize from the master have a stratum of 2, and so on.

The kvpair key "minpoll" indicates the lowest periodic polling that the client will perform against the server. Announcing a large numeric value allows for a server to reduce the amount of NTP messages from clients, but slows down convergence time of clients number of service instances that simultaneously bootstrap.

The kvpair key "key" indicates the NTPv3 authentication mechanism. When present, clients MUST use the value as the key to perform NTPv3 (MD5) hash authentication of message with this service instance. Note that the encryption of the ACP serves as protection of distributing such a cleartext symmetric key via GRASP to clients.

TBD: Understand NTPv4 autokey and define appropriate kvpair to enable auto-configuring it, especially when the service instance announcement indicates the use of the data-plane.

The autoconfiguration described in the following paragraphs is for leafs of the clock distribution graph, e.g., nodes that do only aim to obtain synchronized time from a server. Configuration of the server hierarchy is left to explicit configuration.

Clients SHOULD select service instance(s) with the worst (highest) stratum value. In the face of multiple equal options, clients have to pick the best ones based on the standard selection criteria priority/weight and range, allowing for distributed NTP server deployment by e.e., setting range to 1, or via centralized deployment with multiple servers, setting range to 255 and priority/weight accordingly. Making the stratum the primary selection criteria allows in the future to also introduce autoconfiguration of servers in the NTP clock distribution tree without incurring the problem that a large number of clients would then select higher stratum servers (and overload them).

Like most other autoconfigured services, the autoconfigured NTP time synchronization SHOULD take precedence over explicit configured NTP options to ensure that time synchronization is not subject to misconfiguration of individual nodes (but only subject to misconfiguration of servers).

The kvpair "TZ" option allows to signal the time zone of the ACP network to clients. Its value is a string indicating the time zone of all nodes in the ACP network. Care must be taken not to use this option in networks extending across multiple time zones. Because time zone distribution does not work automatically across larger networks with multiple time zones, overriding the signalled time zone SHOULD be possible through local configuration.

TBD: references for time zone spec standards and also for DST rule indications.

2.3. DNS for operations

Availability of DNS names for network operations/troubleshooting is today mostly an convenience in network operations, but with IPv6 evolving the need to use DNS names even in CLI based network diagnostics is raising - because IPv6 addresses often are more difficult to memorize by operators. More and more network features also support configurition that instead of addresses include domain names or URLs, and ultimately, any non-fully autoconfigured functions should rather rely on domain-names and URLs instead of just addresses for greater flexibility and relilability in the face of address changes.

In thw face of this, ACPna nodes SHOULD support autoconfiguration of DNS for operational purposes. "For operation purposes" implies that the use of of the autoconfigured DNS servers SHOULD NOT be used for DNS services offered to users of the data plane, such as DNS proxy services. This would cause the ACP to effectively carry user traffic, whenever a client DNS request to an ACPna node with a DNS proxy would be forwrdded to an autoconfigured server via the ACP.

The GRASP objective name for such OAM use of DNS is OAM-DNS. It is explicitly not SRV.dns to highlight that this instance of DNS is coped for operational purposes only to isolate it from user issues (performance across the ACP and attacks). Utilizing different DNS contexts also allows to set up split-horizon DNS where all the operationally relevant DNS names are only made available via the DNS servers or zones available across the ACP.

The value of the "search-list" kvpair option is a ";" (semicolon) separated list of domain name prefixes that should be searched by the client for non-FQDN that they need to resolve. "local-arpa" is the prefix to use for reverse IPv4/IPv6 address lookups. If for example "local-arpa" is set to "arpa.example.com", then the clients should first look up IPv4/IPv6 addresses in "ipv6.arpa.example.com."/"in-addr.arpa.example.com." before resorting to lookup in the Internet global "ipv6.arpa."/"in-addr.arpa.". For RFC1918/ULA addresses, no fallback to the global reverse lookup prefixes should be done.

ACPna nodes SHOULD look up their name via a reverse lookup of their ACP address, and then auto-configure this name.

There are no service specifics for the selection of DNS servers. A ACPna node simply uses the standard priority/weight/range options to select a DNS server. It MAY prefer a server with TCP locator-option simply because that allows in most cases faster discovery of connectivity problems than a UDP connection.

TBD: Note that it is fairly easy to re-use the autoconfiguration scheme described here to provide auto-configuration of DNS for user DNS services with the help of the ACP. The objective name would have to be changed and the clocators would have to indicate a data-plane context, so that user requests are carried across the data-plane from DNS proxies to DNS servers. It is unclear if this service should be described in this document though.

2.4. Radius

Radius [RFC2865] is a protocol used for AAA service - Authentication, Authorization and Accounting. Autodiscovery of Radius servers across the ACP for ACPna nodes serves the purpose to enable authentication and authorization of other ACPna autoconfigured services such as below described Section 2.6.

ACPna nodes MUST support Radius and/or Diameter autoconfiguration if they support any of the autoconfigured services depending on such an authentication service.

The GRASP objective name is SRV.radius. The UDP or TCP port of the locator-option in the GRASP header or the clocator option indicate the UDP or TCP port of the Radius servers authentication connection. The context of a clocator MUST be "" to indicate the ACP - because the Radius connections MUST pass across the ACP to be protected against eavesdropping - and the radius security methods described here are not sufficiently secure to allow passing them across the data-plane.

The kvpair "secret_key" string value indicates the secret key to use on the connection to the Radius server. The optional "acct_port" numeric value indicate the UDP/TCP port of any accounting connection supported by the radius server. The protocol (UDP vs. TCP) is the same as the one in the chosen locator-option/clocator.

There are no service specific selection rules. TCP is preferred for faster recognition of a failed server and reselection of an alternative server.

The specific data/authentication/authorization configuration required on the Radius server depends on the OAM service authenticated/authorized and is described in its section in this document.

TBD: Should we define AVpair or different objective names to distinguish what services can be authenticated? Would be easier if we found another service than SSH/Netconf.

2.5. Diameter

TBD. Alternative to Radius. Author would welcome suggesting what parameters are relevant for a diameter authentication service.

2.6. SSH server

ACPna nodes supporting SSH server functionality for remote management access via CLI, NETCONF or other methods SHOULD auto-enable SSH server functionality across the ACP whenever they are aware from ACP GRASP of RADIUS (Section 2.4) or DIAMETER (Section 2.5) authentication servers. ACPna nodes that support ACPna SSH server functionality MUST support authentication via either RADIUS and/or Diameter.

If both protocols are supported by the ACPna node, the ACPna node SHOULD select the authentication server based on the service priority parameters across both protocols. E.g., if a RADIUS server has a higher priority in GRASP than the DIAMETER server, the ACPna node should authenticate against the RADIUS server.

When valid authentication server(s) are discovered, the SSH server is autoconfigured. It SHOULD only listen to the standard SSH port with the ACP address of the node but not be reachable from the data-plane. It MUST NOT be modifiable by configuration (only by auto-configuration). If autoconfiguration of an SSH server on the standard SSH port conflicts with explicitly configured SSH server for the data-plane due to software limitations or complexity, the autoconfigured SSH server MAY be started on a node-type specific and not dynamically selected port number. This port number must be well-known to OAM operations as there is no method provided to signal it to the SSH client side.

Note that this document does not define any standards for the exact message options for authentication or authorization. Especially authorization, such as privilege level that permits to change configuration is likely using vendor specific methods, and Radius/Diameter servers must be capable to recognize the type of client as they had to without this autoconfiguration.

3. Security Considerations

There is no protection against "unauthorized" ACP nodes to generate service announcements, because there is no authorization scheme in GRASP. Discovery of unauthorized announcers is easy though because the service announcements are flooded across the ACP and are therefore easily visible on nodes that may specifically observe announcements to discover unauthorized ones.

A possible framework to define authorization could rely on defining roles for ACP nodes either through additional parameters in their ACP domain certificate or following initial provisioning, and then lock down the ability for later configuration to enable services (and their GRASP announcements) to only those included in the role assigned to the node. This is outside the scope of this document.

4. Change log [RFC Editor: Please remove]

00: Initial version.

5. References

5.1. Normative References

- [I-D.eckert-anima-grasp-dnssd]
Eckert, T., "DNS-SD compatible service discovery in GRASP", draft-eckert-anima-grasp-dnssd-01 (work in progress), July 2018.
- [I-D.ietf-anima-autonomic-control-plane]
Eckert, T., Behringer, M., and S. Bjarnason, "An Autonomic Control Plane (ACP)", draft-ietf-anima-autonomic-control-plane-16 (work in progress), June 2018.
- [I-D.ietf-anima-grasp]
Bormann, C., Carpenter, B., and B. Liu, "A Generic Autonomic Signaling Protocol (GRASP)", draft-ietf-anima-grasp-15 (work in progress), July 2017.
- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, DOI 10.17487/RFC2865, June 2000, <<https://www.rfc-editor.org/info/rfc2865>>.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <<https://www.rfc-editor.org/info/rfc5246>>.
- [RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, DOI 10.17487/RFC5424, March 2009, <<https://www.rfc-editor.org/info/rfc5424>>.
- [RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <<https://www.rfc-editor.org/info/rfc5905>>.

[RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", BCP 165, RFC 6335, DOI 10.17487/RFC6335, August 2011, <<https://www.rfc-editor.org/info/rfc6335>>.

5.2. Informative References

[I-D.ietf-anima-bootstrapping-keyinfra] Pritikin, M., Richardson, M., Behringer, M., Bjarnason, S., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructures (BRSKI)", draft-ietf-anima-bootstrapping-keyinfra-16 (work in progress), June 2018.

Author's Address

Toerless Eckert (editor)
Futurewei Technologies Inc.
2330 Central Expy
Santa Clara 95050
USA

Email: tte+ietf@cs.fau.de