Network File System Version 4
Internet-Draft
Intended status: Informational
Expires: May 27, 2018

S. Hajnoczi, Ed.
Red Hat, Inc.

November 23, 2017

Remote Procedure Call (RPC) over AF_VSOCK
draft-stefanha-nfsv4-rpc-over-vsock-00

Abstract

This document describes how to transfer Remote Procedure Call (RPC) messages over AF_VSOCK sockets. This allows RPC services such as Network File System (NFS) to operate over AF_VSOCK.

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 May 27, 2018.

Table of Contents

1. Introduction

The AF_VSOCK address family facilitates socket communication between a hypervisor and virtual machines running on the hypervisor. It was implemented in Linux 3.9 with initial support for VMware. Further hypervisor support was added for KVM and Hyper-V later. AF_VSOCK services can run on any supported hypervisor because semantics remain the same across hypervisors.

ONC RPC services are bound to transport addresses [RFC1833]. Transport addresses are described by network identifiers (netids) and universal address strings. This standard representation of transport addresses allows address information to be communicated between client programs and RPC services, including the RPCBIND program.

It is necessary to define a network identifier and universal address representation so that ONC RPC may be used over AF_VSOCK. This document describes string representations for the AF_VSOCK netids and universal addresses.

2. Status of AF_VSOCK

This section provides background information that may help reviewers suggest how to proceed with this document. It is not intended to be included unchanged in the final document.

The Linux AF_VSOCK address family does not have a specification document. AF_VSOCK is purely an implementation that first became available in Linux in 2013.

The Linux community generally uses VMware's AF_VSOCK driver as a reference for how sockets should behave. This tends to mirror TCP for SOCK_STREAM and UDP for SOCK_DGRAM so that porting applications from AF_INET to AF_VSOCK is easy.

There is currently momentum towards creating a man page for AF_VSOCK and a shared test suite. This will provide a stronger basis for both users and implementors to work from in the future.

VMware has published reference documentation to their own API, which is called vSockets and encompasses additional header files and APIs not shipped as part of Linux. For the purposes of this document the VMware material is out of scope but might be interesting to those wishing to see a manual on using the vSockets technology that is based on AF_VSOCK: https://code.vmware.com/doc/preview?id=5521.

This raises the question of how this document should reference AF_VSOCK since no external document exists.

3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

4. Intended Use

The intended use of RPC over AF_VSOCK is Network File System (NFS) v4.1 [RFC5661] to export file systems from the hypervisor into virtual machines. It is possible that other RPC services will also be offered by hypervisors to virtual machines.

RPC services can also be provided by virtual machines to the host. Such software is often called a guest agent.

The need for RPC over AF_VSOCK arises because management tools require the ability to deploy RPC services for virtual machines in an automated fashion. Existing TCP/IP network interface configuration, including firewall configuration, is difficult to automate without risk of interfering with the virtual machine owner's networking configuration. AF_VSOCK provides a zero-configuration communications channel with the dedicated purpose of communicating between a hypervisor and virtual machines without these configuration difficulties. This makes it possible to deploy RPC services without requiring manual steps by hypervisor and virtual machine administrators.

5. Network Addresses

AF_VSOCK addresses contain an unsigned 32-bit integer called the Context Identifier (CID) that is the network address. An unsigned 32-bit integer port number acts as the transport selector so that multiple services can be offered on a single network address.

CID values in the range [0, 2] are reserved. Virtual machines are assigned values outside this range. The hypervisor has the well-known CID value 2.

Ports in the range [0, 1023] are privileged and can only be bound by programs that have sufficient privilege.

6. Use of netids for AF_VSOCK

Initial use of RPC over AF_VSOCK is expected to be over connection-oriented sockets since NFS v4.1 [RFC5661] requires reliable, in-order delivery offered by AF_VSOCK connection-oriented sockets. The possibility of using datagram sockets in the future is acknowledged by also reserving a netid for this purpose.

7. Uaddr Format for AF_VSOCK

The format of the uaddr for AF_VSOCK is the US-ASCII string:

```
cid.port
```

The "cid" prefix is the ASCII-decimal representation of the CID network address. The "port" suffix is the ASCII-decimal representation of the port number transport selector.

8. Record Marking on Connection-oriented Sockets

Connection-oriented AF_VSOCK sockets have in-order, guaranteed delivery semantics. RPC messages must be delimited so that message boundaries can be detected. The Record Marking Standard, defined in [RFC5531], is used for this purpose since connection-oriented AF_VSOCK sockets are byte streams.

9. Security Considerations

AF_VSOCK addresses are local to the hypervisor that the virtual machine is running on. Traffic is point-to-point with no routing or forwarding through components external to the hypervisor. This makes AF_VSOCK traffic immune to eavesdropping, replay, message insertion, deletion, modification, and man-in-the-middle attacks as long as the virtual machine and hypervisor are not compromised.

Source address (CID) spoofing by the virtual machine is not possible because the hypervisor enforces the virtual machine's assigned address. RPC services running on the hypervisor MAY use the source CID for authentication since it cannot be faked.

The privileged port range prevents unprivileged processes from binding to low-numbered ports. The privileged port range is a scarce resource but MAY be used to authenticate privileged processes. This allows messages from unprivileged processes to be refused or treated as unathenticated. In particular, RPC AUTH_UNIX authentication MUST only be trusted when the RPC message was sent from a privileged port.

Only AUTH_NULL and AUTH_UNIX flavors are supported although [RFC5531] states that "Standards Track RPC services MUST mandate support for RPCSEC_GSS". Due to the point-to-point nature of AF_VSOCK described above, there are different security considerations than for RPC services deployed on the Internet or even on local networks. RPC services provided by the hypervisor establish the identity of the client by inspecting the source CID and port number. An NFS service facilitates access to an exported file system for which the virtual machine has authorization. The AUTH_UNIX uid and gid are used for NFS file operations but do not give the client access to other virtual machine's file systems. Therefore isolation can be achieved with just AUTH_UNIX.

10. IANA Considerations

10.1. Netids for AF_VSOCK

IANA is asked to assign the following netids from the ONC RPC Netids Registry [RFC5665] on a Standards Action basis:

Netid	Constant Name	RFC(s) and Description	PoC	CR
"vsockc"	NC_VSOCKC	Netid for AF_VSOCK connection-oriented sockets. Section 6 of this document.	IESG	TBD1
"vsockd"	NC_VSOCKD	Netid for AF_VSOCK datagram sockets. Section 6 of this document.	IESG	TBD1

PoC: Point of Contact. CR: Cross Reference to the Uaddr Format Registry.

Table 1: Netids for AF_VSOCK

10.2. Uaddr Format for AF_VSOCK

IANA is asked to assign the following uaddr format from the ONC RPC Uaddr Format Registry [RFC5665] on a Standards Action basis:

Basis	Description and/or Reference	PoC	CR
STDS	Uaddr format for AF_VSOCK. Section 7 of this document.	IESG	TBD1

PoC: Point of Contact. CR: Cross Reference to the Uaddr Format Registry.

Table 2: Uaddr format for AF_VSOCK

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <http://www.rfc-editor.org/info/rfc8174>.

11.2. Informative References

[RFC1833] Srinivasan, R., "Binding Protocols for ONC RPC Version 2", RFC 1833, DOI 10.17487/RFC1833, August 1995, <http://www.rfc-editor.org/info/rfc1833>.

[RFC5531] Thurlow, R., "RPC: Remote Procedure Call Protocol Specification Version 2", RFC 5531, DOI 10.17487/RFC5531, May 2009, <https://www.rfc-editor.org/info/rfc5531>.

[RFC5661] Shepler, S., Ed., Eisler, M., Ed., and D. Noveck, Ed., "Network File System (NFS) Version 4 Minor Version 1 Protocol", RFC 5661, DOI 10.17487/RFC5661, January 2010, <https://www.rfc-editor.org/info/rfc5661>.

[RFC5665] Eisler, M., "IANA Considerations for Remote Procedure Call (RPC) Network Identifiers and Universal Address Formats", RFC 5665, DOI 10.17487/RFC5665, January 2010, <https://www.rfc-editor.org/info/rfc5665>.

Author's Address

Stefan Hajnoczi (editor)
Red Hat, Inc.
100 East Davie Street
Raleigh, NC 27601
USA

Email: stefanha@redhat.com