SECURITY ADVISORY: WIND RIVER TCP/IP STACK (IPNET) VULNERABILITIES

Revision: Version 2.1 25JULY2019

Contact information: Wind River® customers with additional questions about these vulnerabilities should contact Wind River Customer Support or their local Wind River sales representative for more information. If you own a device that may be impacted by these vulnerabilities, please contact your device manufacturer.

Timeline

- 29JUL2019: 3:00 p.m. UTC (8:00 a.m. Pacific Daylight Time) Embargo lifted for each of 11 Common Vulnerabilities and Exposures (CVEs) listed below

Wind River Products Affected

- VxWorks 7 SR540 and VxWorks 7 SR610 ARE affected by one or more of these CVEs (see details below).
- VxWorks 6.9.4.11 and earlier releases are affected by one or more of these CVEs (see details below).
- Older, end-of-life versions of VxWorks, including VxWorks 6.5 and later, are also affected by one or more of these CVEs (see details below).
- All versions of the discontinued product Wind River Advanced Networking Technologies are likely affected by one or more of these CVEs; contact Wind River Customer Support (support@windriver.com) for more details.
- Note 1: Prior to the 2006 Wind River acquisition of Interpeak AB, IPnet was sold as a standalone TCP/IP network stack. Versions of this stand-alone TCP/IP network stack are likely affected by one or more of these CVEs (see details below).
- Note 2: The VxWorks bootrom network stack leverages the same IPnet source as VxWorks and as a result is also technically vulnerable to CVE-2019-12256. The same patches and mitigations apply to VxWorks and the bootrom network stack. However, the bootrom normally uses statically assigned IP addresses, not DHCP. If that is true in your application, then the defects related to those protocols do not apply in practice. Also, a successful exploit of the bootrom network stack has a more difficult timing component. In typical applications, the bootrom does not listen to TCP ports, which means that the TCP-related issues must be timed with the target downloading data from the network.

Wind River Products Not Affected

- The latest release of VxWorks, VxWorks 7 SR620, is NOT affected by any of these CVEs
- VxWorks 5.3 through VxWorks 6.4 inclusive are NOT affected
- VxWorks Cert versions are NOT affected
- VxWorks 653 versions 2.x and earlier are NOT affected
- VxWorks 653 MCE 3.x Cert Edition and later are NOT affected
- VxWorks 653 3.x Multi-core Edition may be affected, contact Wind River Customer Support (support@windriver.com) for more details

Intended Audience

This advisory is intended for your Product Security Incident Response Team (PSIRT), product architects, and software maintenance teams.

Terms of Use

As of the embargo lift date and time listed above, this information is no longer under embargo.

General

Wind River and security researchers have been collaborating on a responsible security disclosure of critical vulnerabilities in the TCP/IP stack used by VxWorks (IPnet). In that time, Wind River has developed and thoroughly tested patches to resolve all of the discovered vulnerabilities. At this time, we have no indication that the discovered vulnerabilities are being exploited in the wild. Nevertheless, if your product uses the IPnet TCP/IP stack, we strongly advise you to apply the patches and release updates to affected devices.

If after reading this security advisory and assessing your use of VxWorks technology, you need to obtain patches, you can obtain the patches in the usual fashion through the Wind River Support Network. Should you require the patches and need technical support on applying the patches, request support via your usual method.

Credit

Ben Seri, Armis Inc. Palo Alto, CA

Identifier

CVE	Defect	Component	CVS Sv3	Title	Interpeak	pre-Vx6.5	Vx 6.5	Vx 6.6	Vx 6.7	Vx 6.8	Vx6. 9.3	Vx6. 9.4	Vx7 pre-SR6 20	Vx7 SR6 20
CVE-2019-12256	V7NET-2423	TCP/IP-stack	9.8	Stack overflow in the parsing of IPv4 packets' IP options	N	N	N	N	N	N	Y	Y	Y	N
CVE-2019-12257	VXW6-87101	DHCP Client	8.8	Heap overflow in DHCP Offer/ACK parsing inside ipdhcpc	N	N	Y	Y	Y	Y	Y	N	N	N
CVE-2019-12255	VXW6-87100	TCP/IP-stack	9.8	TCP Urgent Pointer = 0 leads to integer underflow	Y	N	Y	Y	Y	Y	Y	N	N	N
CVE-2019-12260	V7NET-2425	TCP/IP-stack	9.8	TCP Urgent Pointer state confusion caused by malformed TCP AO option	N	N	N	N	N	N	N	Y	Y	N
CVE-2019-12261	V7NET-2425	TCP/IP-stack	8.8	TCP Urgent Pointer state confusion during connect() to a remote host	N	N	N	N	Y	Y	Y	Y	Y	N
CVE-2019-12263	V7NET-2425	TCP/IP-stack	8.1	TCP Urgent Pointer state confusion due to race condition	N	N	N	Y	Y	Y	Y	Y	Y	N
CVE-2019-12258	V7NET-2426	TCP/IP-stack	7.5	DoS of TCP connection via malformed TCP options	N	N	Y	Y	Y	Y	Y	Y	Y	N
CVE-2019-12259	V7NET-2428	TCP/IP-stack	6.3	DoS via NULL dereference in IGMP parsing	N	N	Y	Y	Y	Y	Y	Y	Y	N
CVE-2019-12262	V7NET-2427	TCP/IP-stack	7.1	Handling of unsolicited Reverse ARP replies (Logical Flaw)	Y	N	Y	Y	Y	Y	Y	Y	Y	N
CVE-2019-12264	V7NET-2428	DHCP Client	7.1	Logical flaw in IPv4 assignment by the ipdhcpc DHCP client	Y	N	Y	Y	Y	Y	Y	Y	Y	N
CVE-2019-12265	V7NET-2428	TCP/IP-stack	5.4	IGMP Information leak via IGMPv3 specific membership report	N	N	Y	Y	Y	Y	Y	Y	Y	N

Remediation

For each issue, applying the patches is STRONGLY recommended and will protect from the vulnerability. For the issues below, alternative mitigations may be applied to your application based on your application and risk assessment. These are work-arounds, not true fixes.

General Remediations

==========

Some system architectures may allow mitigation of these vulnerabilities via firewall settings as detailed below. Where possible, consider immediately enabling these mitigations to protect against an attack while patches are integrated by your team.

Potential Mitigation #1
External Firewall Mitigation

Applicable to: CVE-2019-12255, CVE-2019-12260, CVE-2019-12261, CVE-2019-12263
Applicable to: All affected versions of VxWorks
For applications where devices reside behind a firewall, the four "TCP Urgent Pointer" vulnerabilities can be mitigated via the firewall. Administrators can add a rule to drop/block any TCP-segment where the URG-flag is set. "Urgent data" is a feature that is used by very few applications—it had some uses in the early days of the Internet together with serial terminals, but it is not used by modern applications such as HTTP, SSH, SSL/TLS, etc.

Potential Mitigation #2
VxWorks Firewall Mitigation

Applicable to: CVE-2019-12255, CVE-2019-12260, CVE-2019-12261, CVE-2019-12263
Applicable to: All affected versions of VxWorks

VxWorks has a built-in firewall. Similar to the External Firewall Mitigation, if your application enables the VxWorks firewall, it can be configured to drop/block any TCP segment where the URG flag is set by adding the following rule:

```
'block in quick proto tcp all flags U/U'
```

Potential Mitigation #3
Source Code Mitigation

Applicable to: CVE-2019-12255, CVE-2019-12260, CVE-2019-12261, CVE-2019-12263
Applicable to: All affected versions of VxWorks
A Source Code Mitigation could also do the same thing as the External Firewall Mitigation. As the APIs have stayed the same across versions of IPnet, this mitigation applies to all versions. Describing this mitigation generally:

The first things that happen when enter the TCP-layer are:
1. Verify checksum
2. Extract the TCP header field information, including the flags
3. Process any TCP-options
4. Rest of protocol handing...

The second step *also* extracts the state of the URG flag, which makes it possible to create the following:
if (IP_BIT_ISSET (param.seg.flags_n, IPTCP_FLAG_URG))

```
{
// Drop any TCP segment where the urgent flag is set
ipcom_pkt_free (pkt);
return NULL;
}
```

For applications that can leverage this mitigation, developers can add the above code snippet just before socket lookup (search for a comment that reads "Try to find a matching socket" in all versions in 'iptcp_input()').

Specific Remediations by CVE
==========

CVE-2019-12256
CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Vulnerability EXISTS only for VxWorks 6.9.3 and later. Fixed in Vx7 SR620.
Mitigation: Make call-stack and heap non-executable, which restricts the effect to potential code execution via return-based programming or a DoS attack.
Fix: Patch is required to any system with IPv4 enabled.

Further mitigations:

- Can be mitigated via a firewall by rejecting any IP datagram that contains any of the IP options SSRR (Strict Source and Record Route) or LSRR (Loose Source and Record Route).

The VxWorks firewall cannot select on specific IP options, but it is possible to block packets that carry any IP option. OK in most cases, as use of IP options is uncommon (but there are cases where options are needed).

To drop all packets with IP options, add this rule:

'block in quick all with ipopts'

The rules may be added in a file and consumed, added via the firewall API, or added via the firewall shell-command.

This is how one would add the rules via the shell command:
[vxWorks]# ipf block in quick proto tcp all flags U/U
[vxWorks]# ipf block in quick all with ipopts
Those two commands could be run by a VxWorks boot-script.

CVE-2019-12257
CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Vulnerability EXISTS for only VxWorks 6.6 through VxWorks 6.9.3.
Mitigation: Make call-stack and heap non-executable, which restricts the effect to potential code execution via return-based programming or a DoS attack.
Fix: Patch is required to any system using the DHCP client.

CVE-2019-12255
CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Vulnerability EXISTS for only VxWorks 6.5 through VxWorks 6.9.3.
Vulnerability EXISTS in stand-alone Interpeak IPTCP r6_0_0 and later
Mitigation: Make call-stack and heap non-executable to turn-off remote code execution into DoS.
Mitigation: Enable RFC1122 compatible processing of urgent data.
Fix: Patch to the network stack is required to prevent the attack vector.

CVE-2019-12260
CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Vulnerability EXISTS only for VxWorks 6.9.4.4 and later. Fixed in Vx7 SR620.

Mitigation: Make call-stack and heap non-executable, which restricts the effect to potential code execution via return-based programming or a DoS attack.
Fix: Patch to the network stack is required to prevent the attack vector.

CVE-2019-12261
CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H
Vulnerability EXISTS only for VxWorks 6.7 and later.
Mitigation: Make call-stack and heap non-executable, which restricts the effect to potential code execution via return-based programming or a DoS attack. Fixed in Vx7 SR620.
Fix: Patch to the network stack is required to prevent the attack vector.

CVE-2019-12263
CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H
Vulnerability EXISTS only for VxWorks 6.6 and later. Fixed in Vx7 SR620.
Mitigation: Make call-stack and heap non-executable, which restricts the effect to potential code execution via return-based programming or a DoS attack.
Fix: Patch to the network stack is required to prevent the attack vector.

CVE-2019-12258
CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H
CVE-2019-12259
CVSS:3.0/AV:A/AC:L/PR:N/UI:R/S:U/C:L/I:N/A:H
Vulnerabilities EXIST only for VxWorks 6.5 and later. Fixed in Vx7 SR620.
Mitigation: None
Fix: A patch to the TCP/IP stack is required to prevent the attack vector.

CVE-2019-12262
CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:H
Vulnerability EXISTS only for VxWorks 6.5 and later. Fixed in Vx7 SR620.
Vulnerability EXISTS in stand-alone Interpeak IPNET2 r2_8_0 and later.
Mitigation: Disable RARP.
Fix: Patch to the network stack is required to prevent the attack vector.

CVE-2019-12264
CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:H
Vulnerability EXISTS only for VxWorks 6.5 and later. Fixed in Vx7 SR620.
Vulnerability EXISTS in stand-alone Interpeak IPAPPL r1_2_0 and later.
Mitigation: None
Fix: A patch to the TCP/IP stack is required to prevent the attack vector.

CVE-2019-12265
CVSS:3.0/AV:A/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:L
Vulnerability EXISTS only for VxWorks 6.5 and later. Fixed in Vx7 SR620.
Mitigation: None
Fix: A patch to the TCP/IP stack is required to prevent the attack vector.

Appendix:
User Impact Guide

Wind River TCP/IP Stack (IPnet) Vulnerabilities

July 2019

Table of contents

Primer on CVSS Severity Score

All CVEs are assigned a score between 0 and 10. The score is defined by something called the Common Vulnerability Scoring System version 3 (CVSSv3).

There are several scores defined by CVSS, and the mandatory score for CVEs is called the base score. The base score is the sum of a number of categories, which make the base-score a rough indication for the overall criticality of the issue. It is not possible to give any precise guidelines as to what a specific numeric value means, beyond that a higher value is generally worse than a lower value.

The actual criticality of a defect may still be very different, depending on surrounding factors. As an example, CVE-2019-12258 and CVE-2019-12263 have fairly similar base scores, 7.5 and 8.1, but they differ quite a lot in possible effect if successfully exploited. The former gets a relatively high score for being easy to exploit, but without large impact. The latter is extremely hard to exploit, but has a potential large impact if the exploit succeeds.

The base score is calculated by grading the vulnerability on a three-level scale in six different areas:

Access vector: How the vulnerability may be exploited

Least severity: Local access to the system is required; i.e., the attacker must be able to somehow login to the system.

Medium severity: Adjacent network access is required; i.e., the attacker must be in the same local area network as the victim device.

Greatest severity: Only general network access is needed; i.e., the attacker may be anywhere on the Internet.

Attack complexity: How hard it is to exploit the vulnerability

Least severity: Very precise timing is required, such as using a race condition; or the attack methods require high visibility.

Medium severity: Assigned to cases that require some cooperation from applications, origin of the attacker etc.
Greatest severity: No special conditions are required for an attack.

Authentication: How many times an attacker must authenticate to the victim system in order to exploit a vulnerability

Least severity: The attacker must authenticate multiple times.

Medium severity: A single authentication is sufficient.

Greatest severity: No authentication is required.

Confidentiality: How the confidentiality of the data in the system is affected

Least severity: It may have no impact on the confidentiality of the data.

Medium severity: The data may be partly disclosed; i.e., some data is still protected.

Greatest severity: The data is completely available to the attacker.

Integrity: What control an attacker gains over the data in the victim system

Least severity: The attacker is not able to modify the data.

Medium severity: Parts of the data may be modified by an attacker.

Greatest severity: Complete loss of integrity means that an attacker might have the capability to modify all data on the victim system.

Availability: The availability of the system under, and after, an attack

Least severity: There might be no change to the availability.

Medium severity: The availability may be reduced, with lower performance or loss of some functions.

Greatest severity: The victim system stops working entirely.

Remote Code Execution

From Wikipedia:

In computer security, **arbitrary code execution** (ACE) is used to describe an attacker's ability to execute arbitrary commands or code on a target machine or in a target process. An **arbitrary code execution vulnerability** is a security flaw in software or hardware allowing arbitrary code execution. A program that is designed to exploit such a vulnerability is called an **arbitrary code execution exploit**. The ability to trigger arbitrary code execution over a network (especially via a wide-area network such as the Internet) is often referred to as **remote code execution** (RCE).

Observed Impact

This section tries to give examples of what the observed impact of the CVEs might be when viewed by the user of the VxWorks IPnet TCP/IP-stack.

CVS-2019-12255

With a prerequisite that the system uses TCP sockets, an attacker can either highjack an existing TCP session and inject bad TCP segments, or establish a new TCP session on any TCP port the victim system listens to.

The impact of the vulnerability is a buffer overflow of up to a full TCP receive-windows (by default 10k-64k depending on the version). The buffer overflow happens in the task calling `recv()/recvfrom()/recvmsg()`.

Applications that pass a buffer equal to or larger than a full TCP window are not susceptible to this attack. Applications passing a stack-allocated variable as buffer are the easiest to exploit.

The most likely outcome is a crash of the application reading from the affected socket. In the worst-case scenario, this vulnerability can potentially lead to RCE.

CVE-2019-12256

Not affected by user-application code, this vulnerability resides in the IPv4 option parsing and may be triggered by IPv4 packets containing invalid options.

The most likely outcome of triggering this defect is that the *tNet0* task crashes. In the worst-case scenario, this vulnerability can potentially lead to RCE.

CVE-2019-12257

This vulnerability only affects systems that use the included DHCP client. DHCP packets may go past the local area network (LAN) via DHCP relays, but is otherwise confined to the LAN.

The DHCP client may be used in VxWorks and in the bootrom. Bootrom using DHCP/BOOTP is only vulnerable during the boot process, not after the VxWorks image has booted.

This defect may be used to overwrite the heap, which most likely results in a crash later on a task requesting memory from the heap. In the worst-case scenario, this vulnerability can potentially lead to RCE.

CVE-2019-12258

This vulnerability affects established TCP sessions. An attacker who can figure out the source and destination TCP port and IP addresses of a session can inject invalid TCP segments into the flow, causing the TCP session to be reset.

An application will see this as an `ECONNRESET` error message when using the socket after such an attack.

The most likely outcome is a crash of the application reading from the affected socket.

CVE-2019-12259

This vulnerability requires that at least one IPv4 multicast address has been assigned to the target in an incorrect way, i.e., using the API intended for assigning unicast addresses. It is not possible to exploit for multicast addresses added with the proper API, i.e., `setsockopt()`.

An attacker may use CVE-2019-12264 to incorrectly assign a multicast IP address.

An attacker on the same LAN as the victim system may use this vulnerability to cause a NULL pointer de-reference, which most likely will crash the tNet0 task.

CVE-2019-12260

A prerequisite is that the system uses TCP sockets and listens to at least one TCP port.

The impact of the vulnerability is a buffer overflow of up to a full TCP receive-windows (by default 10k-64k depending on version). The buffer overflow happens in the task calling `recv()/recvfrom()/recvmsg()`.

Applications that pass a buffer equal to or larger than a full TCP window are not susceptible to this attack. Applications passing a stack-allocated variable as buffer are the easiest to exploit.

The most likely outcome is a crash of the application reading from the affected socket. In the worst-case scenario, this vulnerability can potentially lead to RCE.

CVE-2019-12261

A prerequisite is that the system uses TCP sockets and the attacker can trigger the target to establish a new TCP connection that the attacker highjacks the traffic of.

The impact of the vulnerability is a buffer overflow of up to a full TCP receive-windows (by default 10k-64k depending on the version). The buffer overflow happens in the task calling `recv()/recvfrom()/recvmsg()`.

Applications that pass a buffer equal to or larger than a full TCP window are not susceptible to this attack. Applications passing a stack-allocated variable as buffer are the easiest to exploit.

The most likely outcome is a crash of the application reading from the affected socket. In the worst-case scenario, this vulnerability can potentially lead to RCE.

CVE-2019-12262

An attacker residing on the LAN can send reverse-ARP responses to the victim system to assign unicast IPv4 addresses to the target.

The action will not cause any direct harm more than increased usage of RAM. However, the vulnerability may indirectly cause a network connectivity issue for the system on the LAN if the assigned IP addresses collide with other machines.

CVE-2019-12263

A prerequisite is that the system uses TCP-sockets, and there is at least one TCP session enabled that an attacker can inject traffic into.

This vulnerability relies on a race condition between the network task (tNet0) and the receiving application. It is essentially impossible to trigger the race on a system with just a single CPU thread enabled and no way to reliably trigger it on SMP targets.

The impact of the vulnerability is a buffer overflow of up to a full TCP receive-windows (by default 10k-64k depending on the version). The buffer overflow happens in the task calling `recv()/recvfrom()/recvmsg()`.

Applications that pass a buffer equal to or larger than a full TCP window are not susceptible to this attack. Applications passing a stack-allocated variable as buffer are the easiest to exploit.

The most likely outcome is a crash of the application reading from the affected socket. In the worst-case scenario, this vulnerability can potentially lead to RCE.

CVE-2019-12264

An attacker residing on the LAN may choose to highjack a DHCP-client session that requests an IPv4 address. The attacker can send a multicast IP address in the DHCP offer/ack message, which the victim system then incorrectly assigns.

This vulnerability is not very useful in isolation, but can be combined with CVE-2019-12259 to create a denial-of-service attack.

CVE-2019-12265

The IGMPv3 reception handler does not expect packets to be spread across multiple IP fragments. A prerequisite for exploiting this vulnerability is that the victim system has at least one IPv4 multicast address assigned. That prerequisite is almost always fulfilled, as all multicast-capable hosts are required to listen to the all-multicast-hosts address, 224.0.0.1.

Attacks against link local multicast addresses, such as 224.0.0.1, allow an attacker on the LAN to make the victim system transmit data to the network that has not been properly set. Specifically, the data transmitted from the network might be information from packets previously received or sent by the network stack.

Testing patches

The patches are accompanied by a C-source file, `netcve.c`, which contains a set of functions intended to run from the VxWorks C-interpreter.

Add the C-file to the VIP and make sure the component INCLUDE_ROUTECMD is included, then rebuilt.

A successfully patched system should be able to run `netCve2019IsPatched()` and get back 1 as return value. A successful run should look like this:

```
-> netCve2019IsPatched
…some output…
Patches for IPNET CVEs seems to be applied
->
```

A non-patched system will most likely crash when executing the program. Use netCveShow() to list CVEs that are tested by the tool

```
-> netCveShow
 0   CVE-2019-12255
 1   CVE-2019-12256
 …
```

The first column shows the ID of the test and the second column show what CVE the test tries to exploit. A single test can be run with netTestOne *ID*; for example, running the test for CVE-2019-12256.

```
-> netTestOne 1
```

Note that a crash more or less guarantees that the patch for the CVE has not been applied correctly. However, running the test with no apparent effect can happen on some systems even if they lack patches and use an affected version. The programs should be fairly reliable, though.