Network Extensions for Modern macOS
Taking the kernel out of Network Kernel Extensions

Jamie Wood, Internet Technologies
Thank you for your feedback.
System Extensions
System Extensions

Packaged inside of your app

Managed by the OS

Easy to develop and debug

Run independently of any user
System Extensions

Packaged inside of your app

Managed by the OS

Easy to develop and debug

Run independently of any user
Content Filter
NetworkExtension Framework
Content Filter
NetworkExtension Framework

App
Content Filter
NetworkExtension Framework

App

NEFilterManager
Content Filter
NetworkExtension Framework

App
NEFilterManager

System Extension

Flow
DNS
TCP
UDP
IP
Interface
Content Filter
NetworkExtension Framework

App
- NEFilterManager

System Extension
- NEFilterDataProvider
- NEFilterSettings

TCP
- UDP
- IP
- Interface
- DNS

Flow
Content Filter
NetworkExtension Framework

App
NEFilterManager

System Extension
NEFilterDataProvider
NEFilterSettings

Flow
DNS
TCP
UDP
IP
Interface
Content Filter
NetworkExtension Framework

App
- NEFilterManager

System Extension
- NEFilterDataProvider
- NEFilterSettings

TCP
UDP
IP
Interface
Flow
DNS
NEW
Content Filter
NetworkExtension Framework

App
- NEFilterManager

System Extension
- NEFilterDataProvider
- NEFilterSettings

Flow
- DNS

Protocols
- TCP
- UDP
- IP
- Interface
Content Filter
NetworkExtension Framework

App
- NEFilterManager

System Extension
- NEFilterDataProvider
- NEFilterSettings
- NEFilterPacketProvider

Flow
- DNS

TCP
UDP
IP
Interface
Content Filter
NetworkExtension Framework

App
NEFilterManager

System Extension
NEFilterDataProvider
NEFilterSettings
NEFilterPacketProvider

Flow
DNS
TCP
UDP
IP
Interface

NEW
Content Filter
NetworkExtension Framework

App
- NEFilterManager

System Extension
- NEFilterDataProvider
- NEFilterSettings
- NEFilterPacketProvider

TCP
UDP
IP
Flow
DNS
Interface

NEW
Content Filter
NetworkExtension Framework

App
NEFilterManager

System Extension
NEFilterDataProvider
NEFilterSettings
NEFilterPacketProvider

Flow
DNS
TCP
UDP
IP
Interface
Transparent Proxy
NetworkExtension Framework
Transparent Proxy
NetworkExtension Framework

App
Transparent Proxy
NetworkExtension Framework

App
NETTransparentProxyManager
Transparent Proxy
NetworkExtension Framework

App
NETTransparentProxyManager

System Extension

Flow
DNS
TCP
UDP
IP
Interface
Transparent Proxy
NetworkExtension Framework

App
- NETTransparentProxyManager

System Extension
- NEAppProxyProvider
- NENetworkRule

Flow
- DNS
- TCP
- UDP
- IP
- Interface
Transparent Proxy
NetworkExtension Framework

App
- NETTransparentProxyManager

System Extension
- NEAppProxyProvider
- NENetworkRule

Flow
- DNS
DNS Proxy
NetworkExtension Framework
DNS Proxy
NetworkExtension Framework

App
NEDNSProxyManager
DNS Proxy
NetworkExtension Framework

App
- NEDNSProxyManager

System Extension

- DNS
- TCP
- UDP
- IP
- Interface

NEW
DNS Proxy
NetworkExtension Framework

App
- NEDNSProxyManager

System Extension
- NEDNSProxyProvider

- DNS
  - TCP
  - UDP
  - IP
  - Interface
DNS Proxy
NetworkExtension Framework

App
NEDNSProxyManager

System Extension
NEDNSProxyProvider

DNS
VPN
NetworkExtension Framework
VPN
NetworkExtension Framework

App
VPN
NetworkExtension Framework

App

NETunnelProviderManager
VPN
NetworkExtension Framework

App

NETunnelProviderManager

System Extension
VPN
NetworkExtension Framework

App
- NETunnelProviderManager

System Extension
- NEPacketTunnelProvider
VPN
NetworkExtension Framework

App
NETunnelProviderManager

System Extension
NEPacketTunnelProvider

Flow
TCP
UDP
IP
utun

DNS
VPN
NetworkExtension Framework

- App
  - NETunnelProviderManager

- System Extension
  - NEPacketTunnelProvider

- Flow
- DNS
- TCP
- UDP
- IP
- utun
VPN

NetworkExtension Framework

includeAllNetworks

excludeLocalNetworks
Per-App VPN
NetworkExtension Framework

MailDomains
CalendarDomains
ContactsDomains
Custom Protocols

Virtual Machine

Content Filter

Transparent Proxy

DNS Proxy

VPN

Virtual Machine

Custom Protocols
Virtual Machine

vmnet.framework

Shared Mode enhancements  NEW
• IPv6
• Specify IP range of inside network
• Port Forwarding

Bridged Mode  NEW
Custom Protocols

Content Filter

Transparent Proxy

DNS Proxy

VPN

Virtual Machine

Custom Protocols
Custom IP Protocol
Network Framework
Custom IP Protocol
Network Framework
Custom IP Protocol
Network Framework
Custom IP Protocol

Network Framework

Introducing Network.framework

WWDC 2018
let parameters = NWParameters(customIPProtocolNumber: 123)

let destination = NWEndpoint.hostPort(host: "fe80::70", port: 0)
let connection = NWConnection(to: destination, using: parameters)

connection.start(queue: .main)
connection.send(content: packet, completion: .contentProcessed({ error in
    // Check for errors
}))
let parameters = NWParameters(customIPProtocolNumber: 123)

let destination = NWEndpoint.hostPort(host: "fe80::70", port: 0)
let connection = NWConnection(to: destination, using: parameters)

connection.start(queue: .main)
connection.send(content: packet, completion: .contentProcessed({ error in
    // Check for errors
}))
let parameters = NWParameters(customIPProtocolNumber: 123)

let destination = NWEndpoint.hostPort(host: "fe80::70", port: 0)
let connection = NWConnection(to: destination, using: parameters)

collection.start(queue: .main)
collection.send(content: packet, completion: .contentProcessed({ error in
    // Check for errors
}))
```swift
let parameters = NWParameters(customIPProtocolNumber: 123)

let destination = NWEndpoint.hostPort(host: "fe80::70", port: 0)
let connection = NWConnection(to: destination, using: parameters)

connection.start(queue: .main)
connection.send(content: packet, completion: .contentProcessed({ error in
    // Check for errors
}))
```
Custom Link Layer Protocol
Network Framework
Custom Link Layer Protocol
Network Framework
Custom Link Layer Protocol
Network Framework

App

NW EtherneChannel
Custom Link Layer Protocol

Network Framework

App

NW Ethernet Channel

Ethernet Interface
import Foundation
import Network

let path = NWPathMonitor(requiredInterfaceType: .wiredEthernet).currentPath
guard let interface = path.availableInterfaces.first else {
    // Not connected to ethernet
    fatalError("Not connected to ethernet")
}

let channel = NWEthernetChannel(on: interface, etherType: 0xB26E)
import Foundation
import Network

let path = NWPathMonitor(requiredInterfaceType: .wiredEthernet).currentPath

guard let interface = path.availableInterfaces.first else {
    // Not connected to ethernet
    fatalError("Not connected to ethernet")
}

let channel = NWEthernetChannel(on: interface, etherType: 0xB26E)
import Foundation
import Network

let path = NWPathMonitor(requiredInterfaceType: .wiredEthernet).currentPath
guard let interface = path.availableInterfaces.first else {
    // Not connected to ethernet
    fatalError("Not connected to ethernet")
}

let channel = NWEthernetChannel(on: interface, etherType: 0xB26E)
channel.stateUpdateHandler = { state in
    switch state {
    case .ready:
        let packet = Data("Hello".utf8)
        channel.send(content: packet, to: remoteAddress!, vlanTag: 0) { error in
            // Handle error
        }
    default:
        break
    }
}

channel.receiveHandler = { packetData, vlanTag, localAddress, remoteAddress in
    // Handle packet
}

channel.start(queue: .main)
channel.stateUpdateHandler = { state in
    switch state {
        case .ready:
            let packet = Data("Hello".utf8)
            channel.send(content: packet, to: remoteAddress!, vlanTag: 0) { error in
                // Handle error
            }
        default:
            break
    }
}

channel.receiveHandler = { packetData, vlanTag, localAddress, remoteAddress in
    // Handle packet
}

channel.start(queue: .main)
Custom Protocols

Virtual Machine

Content Filter

Transparent Proxy

DNS Proxy

VPN

Virtual Machine

Custom Protocols
Network Kernel Extensions: Endgame

Hard to develop
Hard to debug
Stability problems
Network Kernel Extensions are deprecated.
Summary

- New APIs for Network Extension apps
  - Content filter
  - Transparent proxy
  - DNS proxy
  - VPN
  - Virtual machine
  - Custom protocols

- Network Kernel Extensions are deprecated
### More Information

developer.apple.com/wwdc19/714

<table>
<thead>
<tr>
<th>Event</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Networking Lab</td>
<td>Friday, 9:00</td>
</tr>
<tr>
<td>System Extensions and DriverKit</td>
<td>WWDC 2019</td>
</tr>
<tr>
<td>What’s New in Network Extension and VPN</td>
<td>WWDC 2015</td>
</tr>
</tbody>
</table>