Open Standards

Sumner Evans and Robby Zampino

July 21, 2020

Mines Linux Users Group

What is a Standard?

What is a Standard?

A technical standard is an established norm or requirement for a repeatable technical task. It is usually a formal document that establishes uniform engineering or technical criteria, methods, processes, and practices.[1]

A de facto standard is a custom or convention that has achieved a dominant position by public acceptance or market forces (for example, by early entrance to the market).[2]

[1] https://en.wikipedia.org/wiki/Technical_standard

[2] https://en.wikipedia.org/wiki/De_facto_standard

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What can a standard specify?

Technical standards can apply to all sort of things.

- Network protocols
- Filesystems
- File formats
- Peripherials
- APIs
- Connectors

What is an *Open* Standard?

What is an *Open* Standard?

An open standard is a standard is which a description of the standard is available publically. Open standards frequently also have open development processes, where anyone from the community can help shape the development of the standard.

Who defines Open Standards?

There are a ton of huge organizations that create open standards. The organizations handle definition, publicization, and maintenance

- IEEE
  - 802.11 Wi-Fi
  - 802.3 Ethernet
- IETF
  - RFC793 TCP/IP
  - RFC768 UDP
  - RFC821 SMTP

Who defines Open Standards?

There are a ton of huge organizations that create open standards. The organizations handle definition, publicization, and maintenance

- IEEE
  - 802.11 Wi-Fi
  - 802.3 Ethernet
- IETF
  - RFC793 TCP/IP
  - RFC768 UDP
  - RFC821 SMTP

Who defines Open Standards?

There are a ton of huge organizations that create open standards. The organizations handle definition, publicization, and maintenance

- IEEE
  - 802.11 Wi-Fi
  - 802.3 Ethernet
- IETF
  - RFC793 TCP/IP
  - RFC768 UDP
  - RFC821 SMTP

Who defines Open Standards?

There are a ton of huge organizations that create open standards. The organizations handle definition, publicization, and maintenance

- IEEE
  - 802.11 Wi-Fi
  - 802.3 Ethernet
- IETF
  - RFC793 TCP/IP
  - RFC768 UDP
  - RFC821 SMTP

Who defines Open Standards?

There are a ton of huge organizations that create open standards. The organizations handle definition, publicization, and maintenance

- IEEE
    - 802.11 Wi-Fi
    - 802.3 Ethernet
- IETF
    - RFC793 TCP/IP
    - RFC768 UDP
    - RFC821 SMTP

Who defines *Open* Standards?

There are also smaller organizations that focus on a single area.

- USB-IF
    - Founded in 1983 by Compaq, Digital, IBM, Intel, Microsoft, NEC and Nortel
    - Focused on maintaining and advocating for USB related technologies
- VESA
    - DisplayPort
    - Flat Display Mounting Interface (FDMI)

Who defines *Open* Standards?

There are also smaller organizations that focus on a single area.

- USB-IF
  - Founded in 1983 by Compaq, Digital, IBM, Intel, Microsoft, NEC and Nortel
  - Focused on maintaining and advocating for USB related technologies
- VESA
  - DisplayPort
  - Flat Display Mounting Interface (FDMI)

Who defines *Open* Standards?

There are also smaller organizations that focus on a single area.

- USB-IF
  - Founded in 1983 by Compaq, Digital, IBM, Intel, Microsoft, NEC and Nortel
  - Focused on maintaining and advocating for USB related technologies
- VESA
  - DisplayPort
  - Flat Display Mounting Interface (FDMI)

Who defines *Open* Standards?

There are also smaller organizations that focus on a single area.

- USB-IF
  - Founded in 1983 by Compaq, Digital, IBM, Intel, Microsoft, NEC and Nortel
  - Focused on maintaining and advocating for USB related technologies
- VESA
  - DisplayPort
  - Flat Display Mounting Interface (FDMI)

Who defines *Open* Standards?

There are also smaller organizations that focus on a single area.

- USB-IF
    - Founded in 1983 by Compaq, Digital, IBM, Intel, Microsoft, NEC and Nortel
    - Focused on maintaining and advocating for USB related technologies
- VESA
    - DisplayPort
    - Flat Display Mounting Interface (FDMI)

Why Do We Want Open Protocols?

Open protocols allow

- open development
- clean room implementation
- encourages competition
- auditability

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

Examples of Closed Standards

- Skype
- Whatsapp (based on XMPP)
- Chromecast
- Slack
- Lightning Cables
- HDMI
- CUDA

A Few Terms

There are three main categories of architectures into which most applications that are not running on a single machine can be placed.

1. **Client-server**: also known as *centralized* or *master-slave*, all producers and consumers connect to a single central server. (Example: GitLab)
2. **Decentralized**: the producer and consumer connect directly to one another. (Example: BitTorrent)
3. **Federated**: producers and consumers connect servers, and the servers connect to one another in a decentralized manner. (Example: Matrix)

Each can use open protocols, however federated and decentralized systems allow individuals or groups to own their own infrastructure and still connect to the broader userbase.

A Few Terms

There are three main categories of architectures into which most applications that are not running on a single machine can be placed.

1. **Client-server**: also known as *centralized* or *master-slave*, all producers and consumers connect to a single central server. (Example: GitLab)
2. **Decentralized**: the producer and consumer connect directly to one another. (Example: BitTorrent)
3. **Federated**: producers and consumers connect servers, and the servers connect to one another in a decentralized manner. (Example: Matrix)

Each can use open protocols, however federated and decentralized systems allow individuals or groups to own their own infrastructure and still connect to the broader userbase.

A Few Terms

There are three main categories of architectures into which most applications that are not running on a single machine can be placed.

1. **Client-server**: also known as *centralized* or *master-slave*, all producers and consumers connect to a single central server. (Example: GitLab)
2. **Decentralized**: the producer and consumer connect directly to one another. (Example: BitTorrent)
3. **Federated**: producers and consumers connect servers, and the servers connect to one another in a decentralized manner. (Example: Matrix)

Each can use open protocols, however federated and decentralized systems allow individuals or groups to own their own infrastructure and still connect to the broader userbase.

Cryptographic Standards

Never Roll your own Cryptography

Tons of researchers and penetration testers have already verified the cryptographic security of all of the major cryptographic algorithms.

Open standards like the Advanced Encryption Standard (AES), RSA, OpenPGP, Elliptic Curve DSA, and all of the common hash functions underpin the security of the modern internet.

If these were not open standards, they would not be nearly as well widespread or trusted.

Never Roll your own Cryptography

Tons of researchers and penetration testers have already verified the cryptographic security of all of the major cryptographic algorithms.

Open standards like the Advanced Encryption Standard (AES), RSA, OpenPGP, Elliptic Curve DSA, and all of the common hash functions underpin the security of the modern internet.

If these were not open standards, they would not be nearly as well widespread or trusted.

Never Roll your own Cryptography

Tons of researchers and penetration testers have already verified the cryptographic security of all of the major cryptographic algorithms.

Open standards like the Advanced Encryption Standard (AES), RSA, OpenPGP, Elliptic Curve DSA, and all of the common hash functions underpin the security of the modern internet.

If these were not open standards, they would not be nearly as well widespread or trusted.

The Signal Protocol

The Signal Protocol is a protocol developed by Open Whisper Systems for the Signal messenger.

Many closed-source programs including Facebook Messenger and WhatsApp *claim* to have implemented this same protocol in their instant messenger applications.

The protocol is a combination of well-known cryptographic algorithms including triple Elliptic-Curve Diffie-Hellman and a *Double Ratchet Algorithm*.

Due to the double ratchet, if an attacker cracks the key used to encrypt a single message, they will not be able to decipher subsequent or previous messages.

The Signal Protocol

The Signal Protocol is a protocol developed by Open Whisper Systems for the Signal messenger.

Many closed-source programs including Facebook Messenger and WhatsApp *claim* to have implemented this same protocol in their instant messenger applications.

The protocol is a combination of well-known cryptographic algorithms including triple Elliptic-Curve Diffie-Hellman and a *Double Ratchet Algorithm*.

Due to the double ratchet, if an attacker cracks the key used to encrypt a single message, they will not be able to decipher subsequent or previous messages.

The Signal Protocol

The Signal Protocol is a protocol developed by Open Whisper Systems for the Signal messenger.

Many closed-source programs including Facebook Messenger and WhatsApp *claim* to have implemented this same protocol in their instant messenger applications.

The protocol is a combination of well-known cryptographic algorithms including triple Elliptic-Curve Diffie-Hellman and a *Double Ratchet Algorithm*.

Due to the double ratchet, if an attacker cracks the key used to encrypt a single message, they will not be able to decipher subsequent or previous messages.

The Signal Protocol

The Signal Protocol is a protocol developed by Open Whisper Systems for the Signal messenger.

Many closed-source programs including Facebook Messenger and WhatsApp *claim* to have implemented this same protocol in their instant messenger applications.

The protocol is a combination of well-known cryptographic algorithms including triple Elliptic-Curve Diffie-Hellman and a *Double Ratchet Algorithm*.

Due to the double ratchet, if an attacker cracks the key used to encrypt a single message, they will not be able to decipher subsequent or previous messages.

Application Layer Open Standards in Industry