Internet Draft
Independent Submission
Intended status: Informational
Expires: October 23, 2011
Updates: 1459, 2812, 2813

Richard Hartmann

April 23, 2011

# Default Port for IRC via TLS/SSL
## draft-hartmann-default-port-for-irc-via-tls-ssl-07

Abstract

This document describes the commonly accepted practice of listening on TCP port 6697 for incoming IRC connections encrypted via TLS/SSL.

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 http://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."

Table of Contents

1. Rationale ..... 3
2. Technical Details ..... 3
2.1. Connection Establishment ..... 3
2.2. Why Not STARTTLS ..... 3
2.3. Certiticate Details ..... 4
2.3.1. Server Certificate ..... 4
2.3.2. Client Certificate ..... 4
3. Security Considerations ..... 4
4. IANA Considerations ..... 5
5. Informative References ..... 5
5. Normative References ..... 5
6. Acknowledgements ..... 6

## 1. Rationale

Although system port assignments for both plain text (TCP/UDP port 194) and TLS/SSL [RFC5246] encrypted (TCP/UDP port 994) IRC traffic exist [IANALIST], it is common practice amongst IRC networks not to use them for reasons of convenience and general availability on systems where no root access is granted or desired.

IRC networks have defaulted to listening on TCP port 6667 for plain text connections for considerable time, now. This is covered by the IRCU assignment of TCP/UDP ports 6665-6669.

Similar consensus has been reached within the IRC community about listening on TCP port 6697 for incoming IRC connections encrypted via TLS/SSL.

## 2. Technical Details

### 2.1. Connection Establishment

An IRC client connects to an IRC server. Immediately after that, a normal TLS/SSL handshake takes place. Once the TLS/SSL connection has been established, a normal IRC connection is established via the tunnel. Optionally, the IRC server may set a specific umode for the client, marking it as using TLS/SSL. Again optionally, an IRC server might offer the option to create channels in such a way that only clients connected via TLS/SSL may join.

For details on how IRC works, see [RFC1459], [RFC2810], [RFC2811], [RFC2812], [RFC2813]. Please note that IRC is extremely fragmented and implementation details can vary wildly. Most implementations regard the latter RFCs as suggestions, not as binding.

### 2.2. Why Not STARTTLS

Due to the highly asynchronous nature of IRC, everything other than suspending the whole connection, running STARTTLS and resuming the connection wouldn't work. As there is no concept of suspended connections in IRC, this would require significant effort on the server side and effort on the client side, as well.

The general consensus is that STARTTLS is not worth the effort and that no one is willing to implement it.

While newer protocols can easily design for STARTTLS from the start, IRC is a legacy protocol; implementing STARTTLS in IRC is not realistic.

### 2.3. Certiticate Details

#### 2.3.1. Server Certificate

The IRC server's certificate should be issued by a commonly trusted CA.

The Common Name should match the FQDN of the IRC server or have appropriate wildcards, if applicable.

The IRC client should verify the certificate.

#### 2.3.2. Client Certificate

If the client is using a certificate as well, it should be issued by a commonly trusted CA or a CA designated by the IRC network.

The certificate's Common Name should match the main IRC nickname.

If the network offers nick registration, this nick should be used.

If the network offers grouped nicks, the main nick or account name should be used.

If the network offers nick registration, the client certificate should be used to identify the user against the nick database. See [CERTFP] for a possible implementation.

## 3. Security Considerations

The lack of a common, well established listening port for IRC via TLS/SSL could lead to end users being unaware of their IRC network of choice supporting TLS/SSL. Thus, they might not use encryption even if they wanted to.

It should be noted that this document merely describes client-to-server encryption. There are still other attack vectors like malicious administrators, compromised servers, insecure server-to-server communication, channels that do not enforce encryption for all channel members, malicious clients or comprised client machines on which logs are stored.

Those attacks can by their very nature not be addressed by client-to-server encryption. Additional safe-guards are needed if a user fears any of the threats above.

This document does not address server links as there are no commonly accepted ports or even back-end protocols. Ports and back-end

protocols are normally established in a bilateral agreement. All operators are encouraged to use strong encryption for back-end traffic, no matter if they offer IRC via TLS/SSL to end users.

## 4. IANA Considerations

An assignment of TCP port 6697 for IRC via TLS/SSL will be requested. The proposed keyword is "ircs-u" and the description "Internet Relay Chat via TLS/SSL":

```
ircs-u  6697/tcp       Internet Relay Chat via TLS/SSL
```

Additionally, a clean-up of the current naming scheme will be requested. The author is trying to get into contact with the respective assignees.

```
ircs    994/tcp        Internet Relay Chat via TLS/SSL
ircs    994/udp        Internet Relay Chat via TLS/SSL

irc-u1  6665/tcp       Internet Relay Chat
irc-u2  6666/tcp       Internet Relay Chat
irc-u3  6667/tcp       Internet Relay Chat
irc-u4  6668/tcp       Internet Relay Chat
irc-u5  6669/tcp       Internet Relay Chat
irc-u1  6665/udp       Internet Relay Chat
irc-u2  6666/udp       Internet Relay Chat
irc-u3  6667/udp       Internet Relay Chat
irc-u4  6668/udp       Internet Relay Chat
irc-u5  6669/udp       Internet Relay Chat
```

## 5. Informative References

[IANALIST] http://www.iana.org/assignments/port-numbers , Sep 15, 2010

[TOP100] http://irc.netsplit.de/networks/top100.php , Sep 15, 2010

[MAVERICK] http://irc.netsplit.de/networks/lists.php?query=maverick , Sep 27, 2010

[CERTFP] http://www.oftc.net/oftc/NickServ/CertFP , Mar 17 2011

## 5. Normative References

[RFC1459] J. Oikarinen, Internet Relay Chat Protocol

[RFC2810] C. Kalt, Internet Relay Chat: Architecture

[RFC2811] C. Kalt, Internet Relay Chat: Channel Management

[RFC2812] C. Kalt, Internet Relay Chat: Client Protocol

[RFC2813] C. Kalt, Internet Relay Chat: Server Protocol

[RFC5246] T. Dierks, E. Rescorla, The Transport Layer Security (TLS) Protocol Version 1

## 6. Acknowledgements

Thanks go to the IRC community at large for reaching a consensus.

Special thanks go to the IRC operators who were eager to support port 6697 on their respective networks.

Special thanks also go to Nevil Brownlee and James Schaad for working on this document in their capacities as RFC Editor and Reviewer, respectively.

## APPENDIX A: Supporting data

As of October 2010, out of the top twenty IRC networks [TOP100],[MAVERICK], ten support TLS/SSL. Only one of those networks does not support TLS/SSL via port 6697 and has no plans to support it. All others supported it already or are supporting it since being contacted by the author. A more detailed analysis is available but does not fit within the scope of this document.

## Authors' Address

Richard Hartmann
Munich
Germany
Email: richih.mailinglist@gmail.com
http://richardhartmann.de

## Version History

00 - initial version

01 - fixed [MAVERICK]

02 - removed self-reference as RFC
added reference to [RFC1700]

03 - removed reference to RFC 1700 as per RFC 3232

04 - added section "Technical Details"
expanded section "Security Considerations"
Changed "Intended status" to "Experimental" at RFC authors' suggestion

05 - Moved "Abstract" to the top of the document
Changed "Intended status" back to "Informational"
Added renaming suggestions for old assignments
Removed section "Comments"
Removed ".txt" from document name
Removed "Full Copyright Statement"
Other minor clean-ups

06 - Introduced unique port keys

07 - Extended document based on feedback by James Schaad and Mykyta Yevstifeyev