Reduce Number of Required IPC Parameters

Proposal: 0008
Authors: Operator Foundation
Status: Initial proposal

Implementation: TBD

Introduction

The PT 2.1 specification section on the IPC protocol requires that when launching a transport client all of the common configuration parameters, as well as the client-specific configuration parameters must be set. Similarly, when launching a transport server, all common and server-specific configuration parameters must be set. This proposal seeks to reduce the number of required IPC parameters by making parameters optional when feasible.

Motivation

Launching a transport client or server using the dispatcher is a complex process to learn and understand because of the number of parameters required. However, these are required by the specification and not always actually necessary. By reducing the number of required parameters, use of the dispatcher becomes simplified and so will be easier to learn for new developers.

Proposed solution

All IPC parameters will be marked in the specification as optional except for those which are actually required because no good default parameter value can be chosen.

Design

The sections of the specification which state that all parameters are required will be changed to state that all parameters are optional unless otherwise specified.

1. Common Parameters

1.1 TOR_PT_MANAGED_TRANSPORT_VER or -ptversion

This parameter becomes optional and default to the current version (currently 2.1).

1.2 TOR_PT_STATE_LOCATION or -state

This parameter becomes optional and defaults to the current working directory.

1.3 TOR_PT_EXIT_ON_STDIN_CLOSE or -exit-on-stdin-close

This parameter becomes optional and defaults to 0.

2. Client Parameters

2.1 TOR_PT_CLIENT_TRANSPORTS or -transports

This parameter remains required.

2.2 TOR_PT_PROXY or -proxy

This parameter becomes optional and defaults to being empty.

3. Server Parameters

3.1 TOR_PT_CLIENT_TRANSPORTS or -transports

This parameter remains required.

3.2 TOR_PT_SERVER_TRANSPORT_OPTIONS or -options

This parameter becomes optional and defaults to being empty.

3.3 TOR_PT_SERVER_BINDADDR or -bindaddr

This parameter remains required.

3.4 TOR_PT_ORPORT or -orport on the server or -target on the client

This parameter remains required.

3.5 TOR_PT_EXTENDED_SERVER_PORT or -extorport

This parameter becomes optional and defaults to being empty.

3.6 TOR_PT_AUTH_COOKIE_FILE or -authcookie

This parameter becomes required if -extorport is used, otherwise it is forbidden.

Effect on API Compatibility

These changes only affects the IPC protocol. There is no effect on APIs.

Effect on IPC Compatibility

These changes to the API only loosen requirements and do not add any new requirements. Therefore, this would qualify as a minor change.

Alternatives considered

The alternative is to continue requiring all parameters. However, this is inconsistent within the specification. While the overview of parameters says that they are all required, the language for individual parameters sometimes specifies a default behavior if the parameter is omitted. It would therefore be more consistent to make such parameters optional. If changing the language in the overview, this is a good opportunity for a complete survey and discussion of which parameters should be required and which should be optional, as presented in this proposal.