Cheat Sheet: Building Maltego Integrations

Maltego Integration Concepts

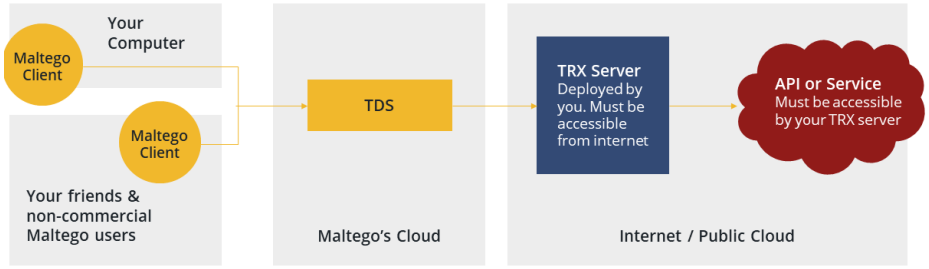
Entities represent things in the world, like IPs, Domains, People, Locations, or anything else.

Transforms allow you to call an API or data source to fetch and connect Entities in Maltego.

Your integration will consist of some Entities that represent the objects in the relevant domain, and some Transforms that make use of the domain's API (or database, service, etc.) to query data to Maltego and/or perform different actions you want to trigger from Maltego.

Architecture

The TRX server (blue box) is your integration code - usually deployed by you as a docker container on a simple cloud VM or application service. The Public TDS is managed by Maltego and allows clients to discover and run your Transforms. You'll need to create an account there and register your Transforms.



TDS Integration

If you're an enterprise developer, the public TDS would be replaced by an iTDS hosted by your org, and everything can be inside your own network.

Helpful Resources

Complete Guide to Building Integrations for Maltego | Maltego Docs: What is a TDS? | System Architecture

High-Level Process

- ☐ Understand the system or API you want to integrate
- ☐ Plan your Maltego Transforms and Entities
- ☐ Start developing Transforms (locally, or remotely right away)
- ☐ Deploy your Transforms and test them in the Maltego ecosystem
- ☐ Launch your integration

Design Phase

Steps to Take

- ☐ Write down objects types / "nouns" (what is represented in your domain?)
- ☐ Sketch relationships between the object types
- ☐ Translate objects into Maltego Entity types
- ☐ Translate relationships to Transforms
- ☐ Make a full list of Transforms you will write, grouped by input and output entity types

Things to Keep it Mind

- ☐ Wherever possible: **Re-use standard Maltego Entity types**
- ☐ Wherever possible: Newly created types should inherit standard types
- ☐ If/when creating entities, use your own namespace (**not maltego.***)
- ☐ **Avoid leaf nodes:** all Entity types should have "outgoing" Transforms
- ☐ Allow **back-and-forth pivots** (every forward Transforms also has an inverse equivalent)

Helpful Resources

Complete Guide to Building Integrations for Maltego | Maltego Standard Entities documentation | Maltego Docs: What is a Transform?

Development Phase

Steps to Take

- ☐ Install Maltego TRX and start a new project
- ☐ (Optional: Run a Transform locally)
- ☐ Implement your Transforms step by step
- ☐ If needed: create your custom Entities)
- ☐ Deploy your TRX server
- ☐ Hook up your Transforms to the Public TDS
- ☐ Iterate on your design and development until your integration is ready
- ☐ If you design your own Entities: Reuse existing types and their properties

Things to Keep it Mind

- ☐ Separate Maltego-logic and API-logic (i.e. write/use a library for the API, and keep Transforms small)
- ☐ Have one "constructor" for each Entity type, don't create/populate them in different places
- ☐ Keep actual Transform code short Separate Maltego-logic and API-logic (i.e. write/use a library for the API, and keep Transforms small)
- ☐ Have one "constructor" for each Entity type, don't create/populate them in different places
- ☐ Keep actual Transform code short

Helpful Resources

Complete Guide to Building Integrations for Maltego

Maltego TRX Library on Github

Maltego Docs:

Setting up and running a local Transform

Setting up and running a Transform on a TDS

Custom Entity creation:

Blog | Docs 1 | Docs 2

Quality Assurance Checklist

- ☐ Typical use-cases of the integration are achievable with the implemented Transforms
- ☐ Compatibility and interoperability with other integrations is ensured
- ☐ Unnecessary complexity is avoided
- ☐ Adherence to the design guidelines
  - Are there any leaf node entities?
  - Do some Transforms "skip" conceptual links?
  - Are any reverse pivots missing?
  - Are link directions and link labels meaningful?
  - Are slider value limits respected?
  - Are namespaces chosen well?
  - Entity re-use and inheritance is well-designed?

Deployment and Launch

- ☐ Deploy your TRX server and configure the Transforms on a TDS (internal or public)
- ☐ Make sure the deployment is production-ready (sufficient workers, reverse proxy if needed, ...)
- ☐ Make sure the deployment is secure (SSL, access restrictions, firewall, ...)
- ☐ If you're planning on commercially offering Transforms to customers, make sure this is covered by your license agreement (reach out to Maltego when in doubt)

If you'd like your integration to be featured on the Maltego Transform hub, reach out via the form on our website.

About Maltego

Maltego empowers investigators worldwide to speed up and increase the precision of their investigations through easy data integration in a single interface, powerful visualization, and collaborative capabilities to quickly zero in on relevant information. With almost one million downloads worldwide since 2008, Maltego is used by a broad audience, from security professionals and pen testers to forensic investigators, investigative journalists, and market researchers.

Learn more about how we can empower your investigations at **www.maltego.com**.