

Projects	Platforms	Language	Description	Status
Common (all platforms)				
Core Framework	Any	C	Handles safe read/writes to and from buffer, chaining; simple debugging framework; useful macros	Active
minisec	Any	C	Our security framework: Generates private/public key pairs with an embedded-friendly cryptographic protocol; Certificates generation and validation; Authentication, key exchange and record-level encryption and authentication; Access rights validation	Active
srtp	Any	C	"Simple Reliable Transport Protocol" - Handles packets retransmission and ordering over BLE/NFC/etc	Active
arpc	Any	C	Client and server RPC framework, handles serialisation, command dispatching and error handling	Active
Common (mobile & server)				
meteora		C++11	Functional programming utilities (observer/observable pattern, filter, map, reduce, etc)	Active
astore		C++14	Mobile/Embedded Linux-friendly datastore, made for efficient replication across devices and robustness	Active
Mobile & Server				

anm-client	iOS, Android, macOS, Linux, Embedded Linux	C++, Objective-C, Java	Main "core" library used by Cordova plugin and daemon	Active
<i>arpc client</i>		C++	Handles services discovery & caching, serialisation from/to JSON and from/to protobuf	Active
<i>DFU</i>		C++	Handles secure delivery of firmware upgrades	Needs updating to v2, planned for Q4 in support of projects
<i>anm-server client</i>		C++	Communicates with the server using GRPC, handles datastore synchronisation, user authentication and requests user certificates for devices access as required	Active
<i>Store</i>		C++	Stores relevant firmware, certificates, data and metadata related to devices and groups	Active
<i>Security</i>		C++	Handles security link (based on minisec), retrieves certificates from store and server as required	Active
<i>BLE</i>		C++	Common cross-platform BLE API	Active
<i>Platform- abstraction layer</i>		C++	Common abstraction for platform-specific features	Active
<i>macOS/iOS implementation</i>	macOS, iOS	Objective-C	macOS and iOS implementation of BLE API & platform-specific features	Active
<i>Android implementation</i>	Android	Java	Android implementation of BLE API & platform-specific features	Needs updating to v2, planned for Q4 in support of projects
<i>Linux implementation</i>	Linux, Embedded Linux	C++	Linux implementation of BLE API & platform-specific features	Needs updating to v2, planned for Q4 in support of projects

anm-daemon	macOS, Linux, Embedded Linux	C++, Objective-C	Desktop/Embedded Linux app based on anm-client providing local APIs Used on desktop platforms as a backend to the JavaScript/TypeScript API	Active
<i>Web Sockets API</i>				Active
<i>GRPC API</i>			Used on desktop platforms mainly for tests (see anm-test below)	In development
anm-cordova-plugin	iOS, Android	Objective-C, Java	Interfaces the anm-client library with the Cordova API for use as a Cordova plugin in cordova projects JavaScript/TypeScript module that connects to the Cordova Plugin (on iOS and Android) or the daemon's Web Sockets API (on macOS and Linux; could potentially be used over an internet connection) and provides the mobile/web app an API for logging in the user, controlling the BLE controller status, discovering BLE devices, connecting to devices with appropriate delegation of rights, executing ARPC commands on the device (in JSON), getting/setting data and metadata	iOS version active, Android version needs updating to v2
anm-typescript	macOS, Linux, iOS, Android	Javascript / TypeScript		Active
anm-server	macOS, Linux	C++	Back-end server Authenticates users of anm-client, and emits certificates for users as required to give them access to a subset of devices with specific access rights for a specific time frame (which means that anm-client does not need to be connected to anm-server permanently)	Active
<i>Security - Users</i>		C++		Active
<i>Security - Devices</i>		C++	Registers devices during provisioning and generates certificates, establishes root of trust, can re-provision devices as required	Reprovisioning feature planned for Q4

<i>Store</i>		C++	Devices belong to a product, hardware version, and groups as defined by the user; users belong to organisations and users groups; firmware and firmware versions are also stored here; data and meta data generated by devices can also be stored here	Active Being updated to v2 (current version), planned for Q4 in support of projects
<i>DFU</i>		C++	Delivers server-signed upgrade images for local upgrading Exposes an API used by anm-client, anm-cli and web apps; using GRPC means we can generate SDKs for any supported language (C++, Java, Obj-C, Python, Ruby, C#, Go, PHP, etc)	Active
<i>GRPC API</i>		C++		Active
anm-server-rest-gw	macOS, Linux	Go	Exposes the GRPC API as a REST API, including Swagger documentation	Active
Firmware				
Firmware SDK	NRF5x	C	Platform implementation of minisec, arpc, srtp; Bluetooth GATT server implementation; Keystore	Active, currently adding NRF52840 support (and take advantage of HW security features) Ongoing migration from v1 to v2 (current version) in support of projects; currently moving all security-features from firmware SDK to bootloader with
Bootloader	NRF5x	C	Platform implementation of minisec, keystore, DFU	

SVC interface + memory
jails/etc

Tools

anm-cli	macOS, Linux	Python	Command-line tool	Active
<i>Provisioning Embedded Code Generator RPC Blob Generator Server Management</i>	macOS, Linux (Host)	Python	Provisioning tool: loads a specific firmware in a new device; recovers the device public key via the debug interface (using PyOCD or J-Link), registers it on the server and loads the device's signed certificate and root of trusts into the device	Active
	macOS, Linux	Python	Generates embedded RPC server code (including error handling, serialisation/deserialisation) for ARPC from Protobuf specs	Active
	macOS, Linux	Python	Generates a Base64 blob describing the RPC interface from Protobuf specs; this blob can then be embedded in a JS/TS app to get the appropriate JS/TS API	Active
	macOS, Linux	Python	CLI interface to the server API	Active
anm-test	macOS, Linux	Python	Python framework for system-wide tests, connects to anm-daemon using GRPC API and can test ARPC APIs on devices	Planned for Q4 (will be used to support projects)

Web/Frontend

Abe	macOS, Linux	Python/Flask	Sample web application which can communicate with anm-server and handle user registration, etc	Requires updating
------------	--------------	--------------	--	-------------------

Bender	macOS, Linux	HTML, TypeScript, Angular	Dashboard, frontend for Abe, user-friendly device management	Experimental
Other/Legacy Projects				
microTLS	mbed/Posix	C	Embedded TLS 1.0/1.1/1.2 client with small footprint and no dynamic memory allocation	Internal Used on some projects for embedded DFU (one chip upgrading another chip)
microOCD	mbed/Posix	C	On-chip debugger for Cortex-M targets, pluggable flash drivers; memory optimised	
flashfs	mbed/Posix	C	Embedded flash system with optional AES encryption/authentication	Internal experiment