

027 – xx Network

network anonymizer – Closeout

8/2/2024

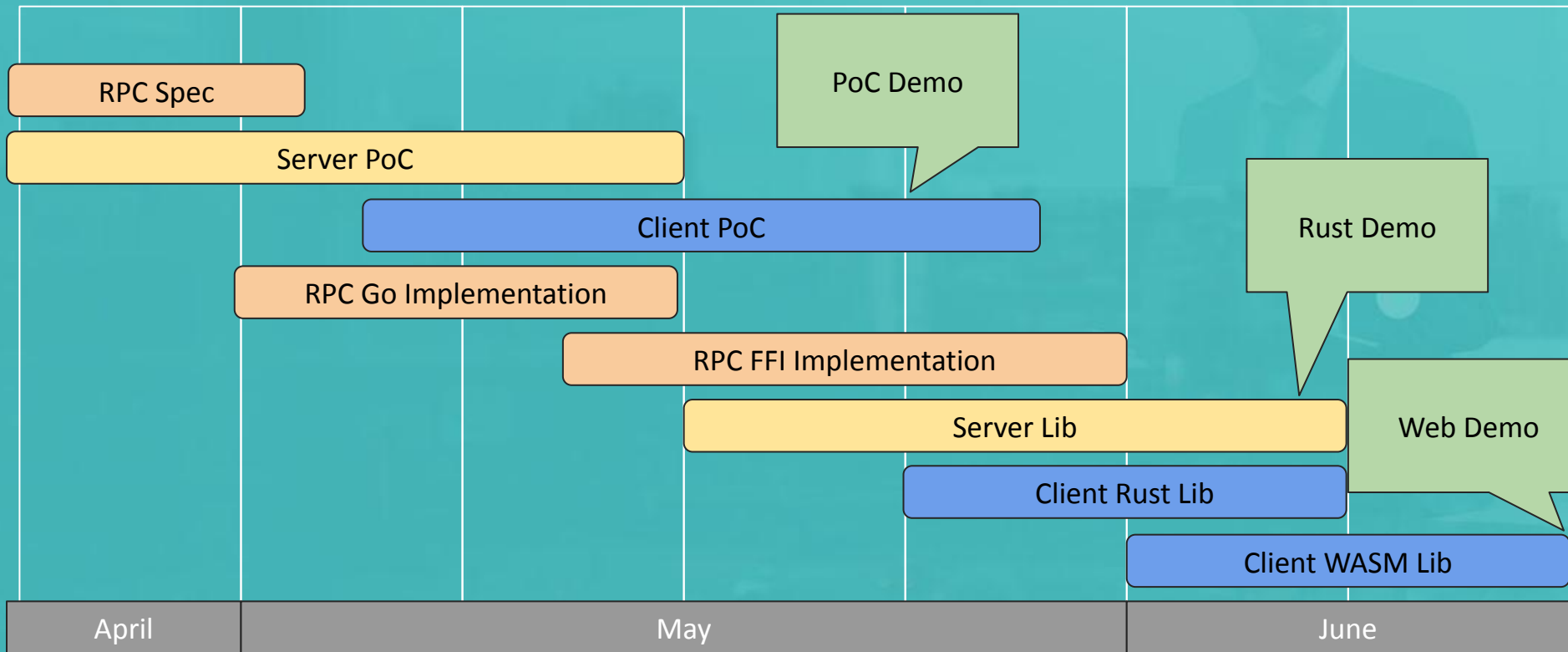
What Was Built

- Simple rust plugin for Worldchain services to communicate over xx network via rust xxdk FFI Implementation
- Client side implementations in Javascript, iOS, Android, golang, C#, and Rust
- Web assembly demo of the working system
- Infrastructure support for the next year

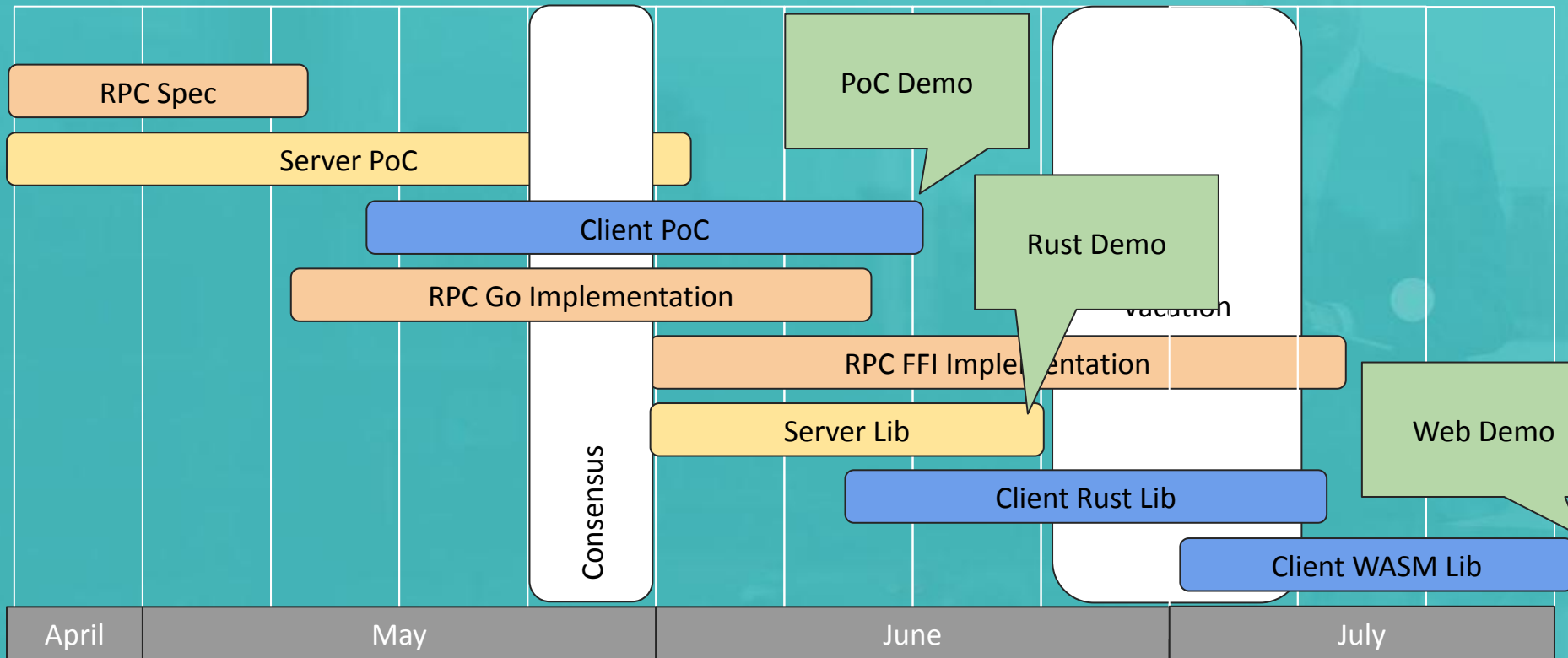
Original Plan

- 1. Discovery Phase: 2 weeks
- 2. Rust xxdk FFI Implementation: Initial phase of 2-3 weeks, with a final phase extending to 6-8 weeks.
- 3. WalletKit Modifications: Starting concurrently with the Rust xxdk implementation, this phase also spans an initial period of 3 weeks and a final period of 6-8 weeks.
- 4. Web Assembly Platform Implementation: Scheduled for the last 3-4 weeks of the project.

Plan



Actual



Rust FFI Lib

- Posted in libxxdk:

📄 23 files +2212 -297

https://git.xx.network/elixxir/libxxdk/-/merge_requests/9

- Integrated with world-tree:

+237 -8

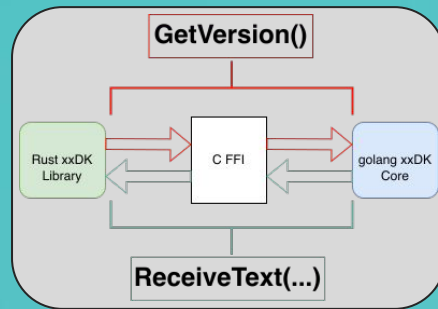
<https://github.com/xxfoundation/world-tree/pull/2>

(note: 120 of those lines are Cargo.lock...)

- Infrastructure templates:

https://git.xx.network/xx_network/worldtree-xx-docker

<https://gitlab.com/carback1/docker-geth>



12 bin/world_tree.rs

```
@@ -16,6 +16,8 @@ use tracing_subscriber::layer::SubscriberExt;
16 16 use tracing_subscriber::util::SubscriberInitExt;
17 17 use world_tree::tree::config::ServiceConfig;
18 18 use world_tree::tree::service::InclusionProofService;
19 + #[cfg(feature = "xxdk")]
20 + use world_tree::tree::service::CmixInclusionProofService;
19 21 + use world_tree::tree::tree_manager::{BridgedTree, CanonicalTree};
20 22 use world_tree::tree::WorldTree;
21 23
@@ -70,10 +72,18 @@ pub async fn main() -> eyre::Result<()> {
70 72
71 73     let world_tree = initialize_world_tree(&config).await?;
72 74
73 -     let handles = InclusionProofService::new(world_tree)
74 +     #[allow(unused_mut)]
75 +     let mut handles = InclusionProofService::new(world_tree.clone())
76 +     .serve(config.socket_address)
77 +     .await?;
78
79
80 +     #[cfg(feature = "xxdk")]
81 +     handles.push(
82 +         CmixInclusionProofService::new(world_tree)
83 +         .serve(config.cmix)
84 +         .await?,
85 +     );
86 +
77 87     let mut handles = handles.into_iter().collect::<FuturesUnordered>
```

82 src/tree/service/cmix.rs

```
17 +
18 + impl<M: Middleware> CmixInclusionProofService<M> {
19 +     pub fn new(world_tree: Arc<WorldTree<M>>) -> Self {
20 +         Self { world_tree }
21 +     }
22 +
23 +     pub async fn serve(
24 +         self,
25 +         cmix_config: RpcServerConfig,
26 +     ) -> eyre::Result<JoinHandle<Result<(), WorldTreeError<M>>>> {
27 +         tracing::info!(?cmix_config, "Initializing cMix RPC server");
28 +
29 +         let router = Router::with_state(self.world_tree)
30 +         .route("inclusionProof", inclusion_proof)
31 +         .route("health", health)
32 +         .route("computeRoot", compute_root);
33 +
34 +         let server_handle = tokio::spawn(async move {
35 +             tracing::info!("Starting cMix RPC server");
36 +             rpc::serve(router, cmix_config)
37 +                 .await
38 +                 .map_err(|e| WorldTreeError::CMixError(e))?;
39 +             Ok(())
40 +         });
41 +
42 +         Ok(server_handle)
43 +     }
44 + }
45 +
```

Client Implementations

- Golang client/v4@v4.7.4:
https://git.xx.network/elixxir/client/-/merge_requests/701
<https://github.com/xxfoundation/elixxir-client>
- Android: <https://central.sonatype.com/artifact/network.xx/android.xxdk>
- iOS: <https://cocoapods.org/pods/XXDK>
- Javascript: <https://www.npmjs.com/package/xxdk-wasm>
- Rust/C#/C: <https://git.xx.network/elixxir/libxxdk>
<https://github.com/xxfoundation/elixxir-libxxdk>
- Examples: https://git.xx.network/xx_network/xxdk-examples
<https://github.com/xxfoundation/xxdk-examples>

Web Assembly Demo

- Live Version:

<https://world-tree.xx.network>

- Source Code:

https://git.xx.network/xx_network/xx-worldtree-web-demo.git

<https://github.com/xxfoundation/xx-worldtree-web-demo>

app > page.tsx > Home

```
1 "use client";
2 import { XXNetwork } from "../xxdk";
3 import IdentityCommitmentForm from "../IdentityCommitmentForm";
4
5 export default function Home() {
6   return (
7     <XXNetwork>
8       <main className="min-h-screen p-10">
9         <IdentityCommitmentForm />
10      </main>
11    </XXNetwork>
12  );
13 }
14
```

app > xxdk.tsx > XXNetwork

```
15 export function XXNetwork({ children }: { children: React.ReactNode })
19   useEffect(() => {
27     xxdk.InitXXDK().then(async (xx: XXDKUtils) => {
30       // Now set up cMix, while other examples download
31       // you must hard code the ndf file for now in your application.
32       const ndf = JSON.stringify(XXNDF);
33
34       // The statePath is a localStorage path that holds cMix xx network state
35       const statePath = "xx";
36
37       // Instantiate a user with the state directory password "Hello"
38       const secret = Buffer.from("Hello");
39       const cMixParamsJSON = Buffer.from("");
40
41       console.log(secret);
42
43       const stateExists = localStorage.getItem("cMixInitialized");
44       if (stateExists === null || !stateExists) {
45         await xx.NewCmix(ndf, statePath, secret, "");
46         localStorage.setItem("cMixInitialized", "true");
47       }
48       xx.LoadCmix(statePath, secret, cMixParamsJSON).then((net: CMix) => {
49         // Once all of our clients are loaded we can start
50         // listening to the network
51         net.StartNetworkFollower(10000);
52         net.WaitForNetwork(30000);
53         setXXCMix(net);
54       });
55     });
56   }, []);
57
58   return (
59     <XXContext.Provider value={XXDKUtils}>
```

WebAssembly Demo

Next Steps?

1. Minor bug will be fixed in next xxdk release
2. Would love to get the PR into world-tree's codebase
 - a. Maintenance is adding a line for new endpoints
3. Can we integrate into a real client? Another Service?



Thank You!

network anonymizer – closeout

7/31/2024