

Hi!

Happy to introduce new functionality in [suave-geth](#) to emit onchain events generated during the offchain computations on the Kettle.

Before, emit events like this were not possible with Suave:

```
contract MySuapp { function onchain() public { }
```

```
event OffchainEvent(); function offchain() public returns (bytes memory) { emit OffchainEvent(); returns  
abi.encodeWithSelector(this.onchain.selector); } }
```

Though the code is valid Solidity, emit OffchainEvent()

was a nil operation, until now.

We [added support](#) to send back to the onchain execution the logs emitted offchain. The Suapp requires minimal changes to support this feature since most of the logic is handled by [suave-std](#):

```
import "suave-std/Suapp.sol";
```

```
contract MySuapp is Suapp { function onchain() emitOffchainLogs public { }
```

```
event OffchainEvent(); function offchain() public returns (bytes memory) { emit OffchainEvent(); returns  
abi.encodeWithSelector(this.onchain.selector); } }
```

You can find a complete example in the [suapp-examples](#) repository.

There are still some open questions like how do you make sure certain confidential information cannot ever be leaked out with this framework.

Looking forward to receiving your feedback on this.