μAFL

提出μAFL——一种fuzz微控制器固件的硬件范围内的方法,它利用现有嵌入式系统开发中的调试工具来构建一个兼容AFL的fuizzng框架。

提出使用ARM ETM做非入侵式的反馈收集，

为了提高效果，μAFL采用LCSAJ（线性代码序列和跳转）分析来直接处理原始ETM数据而不需要花费高额的反编译。实现并评估了原型系统在两种主流MCU的SDK。将一个UBS驱动的作为case study来展示他们的原型系统可以fuzz真实世界的驱动代码。帮助他们发现13个未知bug和8个CVE。

1. 主机将测试用例通过调试狗发送到目标板子的预留内存
2. 通知目标板子开始执行
3. 当目标固件达到第一次使用testcase的指令位置，主机就会发送命令来激活ETM功能
4. 当固件执行时，生成的Trace通过调试狗同步发送到主机
5. 在完成一轮执行后，主机再发送一条命令关闭ETM
6. 将使用收集到的跟踪信息重建Tracce。
7. 最终的结果被映射到位图中，以确定是否发现了新的路径，并按照AFL的相同遗传算法指导新测试用例的生成

