BabyAuth writeup

本题主要考查实际使用椭圆曲线签名以及分组密码CBC工作模式时的部分细节，题中采用了Github上的一个国密算法库 (https://github.com/duanhongyi/gmssl) 实现的国密算法SM2、SM3、SM4。

题目描述

题目服务器每次连接都会生成一串 `session id`，对应的管理员用户名为'admin'||session id。 题目两个功能分别对注册与登录：

- 注册时，用户提供一个用户名，注意不能包含管理员用户名。然后服务器使用生成一个json字符串，其中 `rwx` 置为False、 `id` 置为用户名。接着使用SM2对其签名、SM3对其计算摘要，将三者串联起来用CBC模型的SM4加密得到token，并将token返回给用户。
- 登录时，用户提供一个token，解密后，若摘要与签名均匹配则登录成功，若 `rwx` 为True且 `id` 为管理员用户名，则返回flag。

解题思路

先简述本题中利用的两个核心点：

- https://github.com/duanhongyi/gmssl/blob/master/tests/test_sm2.py#L22，SM2签名时默认调用了 `sign()` 函数，而此函数传入的消息应当为摘要值，本题中的参数为消息明文，假如两个消息m_0, m_1满足$m_0 \equiv m_1 \pmod{n}$，这里n为曲线的阶，那么对m_0的签名显然适用于m_1。
- https://github.com/duanhongyi/gmssl/blob/master/gmssl/func.py#L16， SM4采用的PKCS7 padding在去padding时，没有校验padding合法性（假设最后一个明文字节是x，应当满足x在1-16之间，且最后x个字节的值均为x）。

最终我们肯定要构造出一个SM4的密文，其解密结果包含完整的管理员用户名，但我们又不能在注册时直接使用该名字，为了绕过此限制，考虑将原json头 `{"rwx": False, "` 压缩为 `{"rwx":1,"id":"a` ，这样将管理员用户名除第一个字符外的后续部分作为用户名注册即可拿到对应的SM4密文，接着通过修改IV使得前两个密文解析出的"id"字段恰好为管理员用户名。接着需要构造合法的SM2签名，利用第一个漏洞，我们可以找出另一个合法用户名对应的json恰好和我们目标的json发生了碰撞，即两者模n下相等。而最后的SM3摘要我们可以预先计算出来，注意到其为hex形式，故可以直接作为用户名注册，从而拿到对应的SM4密文。

最后需要解决unpadding问题，上面第二个漏洞使得我们可以移除任意长度的明文块。

签名碰撞

这一步可以通过格归约完成，此处直接给一个简化版的例子。假设，我们需要构造一个字符串形如 `0123456789abcdef????` （其中末4个字节可控且为小写字母，值分别记为c_1, c_2, c_3, c_4；前面若干字节已知，对应的数值记为k），该字符串的值$x := k + c_4 + 2^8 c_3 + 2^{16} c_2 + 2^{24} c_1$满足$x \equiv t \pmod{n}$。

考虑构造格$\mathcal{L}(\mathbf{B})$如下

$$\mathbf{B} := \begin{bmatrix} Cn & & & & & \\ C & 1 & & & & \\ C2^8 & & 1 & & & \\ C2^{16} & & & 1 & & \\ C2^{24} & & & & 1 & \\ C(t-k) & D & D & D & D & 1 \end{bmatrix},$$

其中C是一个足够大的数，D设置为接近小写字母均值110的数，那么很可能存在短向量$\mathbf{v} := (0, c_4 - D, c_3 - D, c_2 - D, c_1 - D, -1) \in \mathcal{L}(\mathbf{B})$，故可以使用LLL或BKZ算法求出$\mathbf{v}$进而构造出满足要求的字符串。

任意unpadding

假设我们有一个合法的token，我们可以在其后面添加两个消息块，构造出新token，即$token' := token||c_1||c_2$，其中$c_2$完全随机生成，而$c_1$除最后一字节外其余均为\x00。我们每次改变$c_1$的最后一个字节，遍历255种可能，则必然存在一个值使得新的token也是合法的，此时相当于我们知道了$Dec(c_2)$的最后一字节。那么通过设置c_1的最后一字节，我们就可以使得$Dec(c_2) \oplus c_1$的最后一字节为指定数值，而实现任意长度的unpadding。这里我们将一组c_1, c_2记为p。

最终构造

那么最后的payload形如$iv'||ct_{name}||p||ct_{sig}||p||ct_{dig}||p||p$，即在每一小部分后面都一个padding块，使得unpadding的结果为预期值。