

A meet-in-the-middle attack on structure seeds, based on the structure of xor M

Assumptions

Let's assume we found a structure and we know its chunk position. Here's the relationship, assuming the [document](#) is true:

$$s_0 = (w + Px_r + Qz_r + salt) \oplus M \pmod{2^{48}}$$

$$x_c = x_r \cdot spacing + (s_1 \gg 17) \bmod bound$$

$$z_c = z_r \cdot spacing + (s_2 \gg 17) \bmod bound$$

I will also claim the following relationships are true:

$$x \gg n = \frac{x - (x \bmod 2^n)}{2^n}$$

$$(m \cdot x) \bmod m \cdot n = m \cdot (x \bmod n)$$

where $m, n \in \mathbb{N}$

If $(a \bmod n) + (b \bmod n) \geq n$:

$$(a + b) \bmod n = (a \bmod n) + (b \bmod n) - n$$

If $(a \bmod n) + (b \bmod n) < n$:

$$(a + b) \bmod n = (a \bmod n) + (b \bmod n)$$

Derivation of "same-value" constraint

Let's rearrange the equations in terms of s_i :

$$s_0 = (w + Px_r + Qz_r + salt) \oplus M \pmod{2^{48}}$$

$$(s_1 \gg 17) \bmod bound = x_c - x_r \cdot spacing$$

$$(s_2 \gg 17) \bmod bound = z_c - z_r \cdot spacing$$

Let's do a bit of renaming here:

- $Px_r + Qz_r + salt = D$
- $x_c - x_r \cdot spacing = \Delta x$
- $z_c - z_r \cdot spacing = \Delta z$

We get:

$$s_0 = (w + D) \oplus M \pmod{2^{48}}$$

$$(s_1 \gg 17) \bmod bound = \Delta x$$

$$(s_2 \gg 17) \bmod bound = \Delta z$$

Notice that $\lfloor \log_2(M) \rfloor = 34$, which means that it affects this amount of low bits plus one. But we still have $48 - 35 = 13$ unaffected high bits. And some other bits are sprinkled here and there in the middle.

Rewriting the first equation to account for that:

$$s_0 = 2^{35} \cdot (w + D) \gg 35 + e_0 \pmod{2^{48}}$$

$$(s_1 \gg 17) \bmod bound = \Delta x$$

$$(s_2 \gg 17) \bmod bound = \Delta z$$

where:

- $e_0 = (w + D) \oplus M \pmod{2^{35}}$

So, at the cost of adding another variable, albeit small, we are able to make the expression slightly more algebraically malleable. We could keep $2^{35} \cdot (w + D) \gg 35$ as it is, but the problem is that this expression varies for each structure. Since we are trying to combine constraints here, we'll have to split it up.

What I know for certain is that we can define bitshifts as such:

$$x \gg n = \frac{x - x \bmod 2^n}{2^n}$$

Also, splitting $(a + b) \bmod 2^n$ into $a \bmod 2^n + b \bmod 2^n$ is possible, the problem is that when we do that, we get a carry condition, which is basically another variable, although it can only take on two states, which is a lot easier to deal with, than a full on variable.

At the same time, D is already known for each structure. Then, after splitting the mod bracket, we'll find ourselves with $w - w \bmod 2^{35}$, which can be effectively called w_h , h stands for "the high bits of w ". Feels like this is a good option to avoid collecting another variable.

With that in mind, let's do a bit of algebra on s_0 :

$$s_0 = 2^{35} \cdot (w + D) \gg 35 + e_0 \pmod{2^{48}}$$

Expanding our bitshift definition:

$$s_0 = w + D - (w + D) \bmod 2^{35} + e_0 \pmod{2^{48}}$$

Now, we know that:

If $w \bmod 2^{35} + D \bmod 2^{35} \geq 2^{35}$:

$$(w + D) \bmod 2^{35} = w \bmod 2^{35} + D \bmod 2^{35} - 2^{35}$$

If $w \bmod 2^{35} + D \bmod 2^{35} < 2^{35}$:

$$(w + D) \bmod 2^{35} = w \bmod 2^{35} + D \bmod 2^{35}$$

Effectively, our expression becomes this:

$$(w + D) \bmod 2^{35} = w \bmod 2^{35} + D \bmod 2^{35} - 2^{35} \cdot c_0$$

where:

- $c_0 \in \{0, 1\}$

Notice that we also will subtract $D \bmod 2^{35}$ from D , which will also leave us with just the higher bits of D . This means that:

$$2^{35} \cdot (w + D) \gg 35 = w + D - w \bmod 2^{35} - D \bmod 2^{35} + c_0 \cdot 2^{35} = 2^{35} \cdot (w \gg 35 + D \gg 35 + c_0)$$

This property extends to arbitrary a and b , so we found a way to give bitshifts almost linear properties. Almost is good enough here.

As discussed earlier, let's rename $w \gg 35$ to w_h and $D \gg 35$ to D_h to reduce clutter.

Let's use our new property, then:

$$s_0 = 2^{35} \cdot (w_h + D_h + c_0) + e_0 \pmod{2^{48}}$$

where:

- $e_0 = (w + D) \oplus M \pmod{2^{35}}$

In other words:

$$2^{35} \cdot w_h = s_0 - 2^{35} \cdot (D_h + c_0) - e_0 \pmod{2^{48}}$$

where:

- $e_0 = (w + D) \oplus M \pmod{2^{35}}$

Now let's look at the other two equations:

$$(s_1 \gg 17) \bmod bound = \Delta x$$

$$(s_2 \gg 17) \bmod bound = \Delta z$$

Let's expand $\bmod bound$:

$$(s_1 \gg 17) = \Delta x + k_1 \cdot bound$$

$$(s_2 \gg 17) = \Delta z + k_2 \cdot bound$$

Substitute in our definitions of s_1 and s_2 :

$$(((s_0 \cdot M + A) \bmod 2^{48}) \gg 17) = \Delta x + k_1 \cdot bound$$

$$(((s_0 \cdot M^2 + A \cdot (M + 1)) \bmod 2^{48}) \gg 17) = \Delta z + k_2 \cdot bound$$

Now, remember the definition of s_0 :

$$s_0 = 2^{35} \cdot (w_h + D_h + c_0) + e_0 \pmod{2^{48}}$$

Let's.. look at the first equation.

$$(((s_0 \cdot M + A) \bmod 2^{48}) \gg 17) = \Delta x + k_1 \cdot bound$$

Let's substitute in our definition of s_0 :

$$((((2^{35} \cdot (w_h + D_h + c_0) + e_0) \cdot M + A) \bmod 2^{48}) \gg 17) = \Delta x + k_1 \cdot bound$$

If we expand the left hand side of our equation using the bitshift formula, we get the following:

$$(((2^{35} \cdot (w_h + D_h + c_0) + e_0) \cdot M + A) \bmod 2^{48}) - (((2^{35} \cdot (w_h + D_h + c_0) + e_0) \cdot M + A) \bmod 2^{17})$$

Say, we wanna extract only w_h , as that's constant across... many equations. We'll be splitting that left bracket into two mod brackets, albeit with additional carry added to the mix. My point is, let's drop only w_h in the second term:

$$(((2^{35} \cdot (w_h + D_h + c_0) + e_0) \cdot M + A) \bmod 2^{48}) - (((2^{35} \cdot (D_h + c_0) + e_0) \cdot M + A) \bmod 2^{17})$$

If we split purely the first bracket, we get:

$$((2^{35} \cdot M \cdot w_h) \bmod 2^{48}) + (((2^{35} \cdot (D_h + c_0) + e_0) \cdot M + A) \bmod 2^{48}) - c_1 \cdot 2^{48}$$

where:

- $c_1 = 1$ if:

$$((2^{35} \cdot M \cdot w_h) \bmod 2^{48}) + (((2^{35} \cdot (D_h + c_0) + e_0) \cdot M + A) \bmod 2^{48}) \geq 2^{48}$$

- $c_1 = 0$ if:

$$((2^{35} \cdot M \cdot w_h) \bmod 2^{48}) + (((2^{35} \cdot (D_h + c_0) + e_0) \cdot M + A) \bmod 2^{48}) < 2^{48}$$

Notice that the terms inside the first bracket match exactly. And we know that:

$$(a \bmod 2^{48}) \bmod 2^{35} = a \bmod 2^{35}$$

Let's label $X = 2^{35} \cdot (D_h + c_0) + e_0$. From this we know that $X \bmod 2^{35} = e_0$

With that, we get:

$$((2^{35} \cdot M \cdot w_h) \bmod 2^{48}) + ((X \cdot M + A) \bmod 2^{48}) - c_1 \cdot 2^{48}$$

where:

- $c_1 = 1$ if:

$$((2^{35} \cdot M \cdot w_h) \bmod 2^{48}) + (X \bmod 2^{48}) \geq 2^{48}$$

- $c_1 = 0$ if:

$$((2^{35} \cdot M \cdot w_h) \bmod 2^{48}) + (X \bmod 2^{48}) < 2^{48}$$

With that, it looks like we can use our definition of a bitshift here. With this in mind, we get:

$$2^{35} \cdot ((M \cdot w_h) \bmod 2^{13}) + 2^{17} \cdot ((X \bmod 2^{48}) \gg 17) - c_1 \cdot 2^{48} = 2^{17} \cdot (\Delta x + k_1 \cdot bound)$$

Divide everything by 2^{17} :

$$2^{18} \cdot ((M \cdot w_h) \bmod 2^{13}) + (((X \cdot M + A) \bmod 2^{48}) \gg 17) - c_1 \cdot 2^{31} = \Delta x + k_1 \cdot bound$$

Leave the constant part on the left:

$$2^{18} \cdot ((M \cdot w_h) \bmod 2^{13}) = \Delta x + k_1 \cdot bound - (((X \cdot M + A) \bmod 2^{48}) \gg 17) + c_1 \cdot 2^{31}$$

For the second equation this relationship becomes:

$$2^{18} \cdot ((M^2 \cdot w_h) \bmod 2^{13}) = \Delta z + k_2 \cdot bound - (((X \cdot M^2 + A \cdot (M + 1)) \bmod 2^{48}) \gg 17) + c_2 \cdot 2^{31}$$

As $k_i \cdot bound$ are defined to be within bounds for $\bmod 2^{31}$, I am pretty sure the divisibility check property may work. With that in mind, let's take the whole expression $\bmod 2^{31}$:

$$2^{18} \cdot ((M \cdot w_h) \bmod 2^{13}) = \Delta x + k_1 \cdot bound - (((X \cdot M + A) \bmod 2^{48}) \gg 17) \pmod{2^{31}}$$

$$2^{18} \cdot ((M^2 \cdot w_h) \bmod 2^{13}) = \Delta z + k_2 \cdot bound - (((X \cdot M^2 + A \cdot (M + 1)) \bmod 2^{48}) \gg 17) \pmod{2^{31}}$$

Now, bring 2^{18} back into the mod bracket:

$$((2^{18} \cdot M \cdot w_h) \bmod 2^{31}) = \Delta x + k_1 \cdot bound - (((X \cdot M + A) \bmod 2^{48}) \gg 17) \pmod{2^{31}}$$

$$((2^{18} \cdot M^2 \cdot w_h) \bmod 2^{31}) = \Delta z + k_2 \cdot bound - (((X \cdot M^2 + A \cdot (M + 1)) \bmod 2^{48}) \gg 17) \pmod{2^{31}}$$

Now, swap the order of mods and bitshifts:

$$((2^{18} \cdot M \cdot w_h) \bmod 2^{31}) = \Delta x + k_1 \cdot bound - (((X \cdot M + A) \gg 17) \bmod 2^{31}) \pmod{2^{31}}$$

$$((2^{18} \cdot M^2 \cdot w_h) \bmod 2^{31}) = \Delta z + k_2 \cdot bound - (((X \cdot M^2 + A \cdot (M + 1)) \gg 17) \bmod 2^{31}) \pmod{2^{31}}$$

Let's drop $\bmod 2^{31}$:

$$2^{18} \cdot M \cdot w_h = \Delta x + k_1 \cdot bound - ((X \cdot M + A) \gg 17) \pmod{2^{31}}$$

$$2^{18} \cdot M^2 \cdot w_h = \Delta z + k_2 \cdot bound - ((X \cdot M^2 + A \cdot (M + 1)) \gg 17) \pmod{2^{31}}$$

Note that $X = 2^{35} \cdot (D_h + c_0) + e_0$, where $e_0 \in [0, 2^{35} - 1]$. If we were to brute force k_i instead, $k_i \in [\lfloor \Delta x / bound \rfloor, \lfloor (\Delta x + 2^{31} - 1) / bound \rfloor]$, resulting in roughly $2^{31} / bound$ values. But the problem here is, to brute force e_0 , you need to also go through all of the "errors" of bitshifting. That's additional 2^{17} values per each k_i search, resulting in roughly $2^{48} / bound$ total seeds to look through.

However, there is a way to lower the search space a bit when brute forcing e_0

mod gcd(2^{18} , bound)

Let's look at the expression $\bmod \gcd(2^{18}, bound)$ For convenience, we'll label $\gcd(2^{18}, bound) = g$. Let's see what happens:

$$0 = \Delta x - (((2^{35} \cdot (D_h + c_0) + e_0) \cdot M + A) \gg 17) \pmod{g}$$

Pretty sure we can also drop 2^{35} here:

$$\Delta x - ((e_0 \cdot M + A) \gg 17) = 0 \pmod{g}$$

- To prove that you can do that, you'd need to expand the bitshift definition, split off the $2^{35} \cdot M \cdot (D_h + c_0)$ term from the mod bracket. Notice that you can move 2^{35} from $\text{mod } 2^{48}$ as 2^{35} divides 2^{48} . But 2^{35} is divisible by $\text{gcd}(2^{48}, \text{bound})$, so that term vanishes.

Let's expand the bitshift:

$$2^{17} \cdot \Delta x - e_0 \cdot M - A + r = 0 \pmod{2^{17} \cdot g}$$

Now let's define e_0 :

$$e_0 \cdot M = 2^{17} \cdot \Delta x - A + r \pmod{2^{17} \cdot g}$$

$$e_0 = M^{-1} \cdot (2^{17} \cdot \Delta x - A + r) \pmod{2^{17} \cdot g}$$

$$e_0 = (M^{-1} \cdot (2^{17} \cdot \Delta x - A + r)) \text{ mod } (2^{17} \cdot g) + q \cdot 2^{17} \cdot g$$

where:

- $r \in [0, 2^{17} - 1]$
- $q \in [0, 2^{18}/g - 1]$

Okay, so... bruteforcing r gives us search space of 2^{17} values. Then, bruteforcing q gives us $2^{35}/(g \cdot 2^{17}) = 2^{18}/g$ values. Combining these, we get the search space of $2^{17} \cdot 2^{18}/g = 2^{35}/g$ values, which is a bit better than what we had before.

Note the original expression:

$$2^{18} \cdot M \cdot w_h = \Delta x + k_1 \cdot \text{bound} - ((X \cdot M + A) \gg 17) \pmod{2^{31}}$$

Leave only $\Delta x + k_1 \cdot \text{bound}$ on the right side:

$$2^{18} \cdot M \cdot w_h + ((X \cdot M + A) \gg 17) = \Delta x + k_1 \cdot \text{bound} \pmod{2^{31}}$$

Apply $\text{mod } 2^{31}$ to both sides separately:

$$(2^{18} \cdot M \cdot w_h + ((X \cdot M + A) \gg 17)) \text{ mod } 2^{31} = (\Delta x + k_1 \cdot \text{bound}) \text{ mod } 2^{31}$$

But $\Delta x + k_1 \cdot \text{bound}$ is defined in such a way to never exceed 2^{31} . So, there's no "wraparound" effect happening. Because of that, we can just drop the mod on the right hand side:

$$(2^{18} \cdot M \cdot w_h + ((X \cdot M + A) \gg 17)) \text{ mod } 2^{31} = \Delta x + k_1 \cdot \text{bound}$$

This tells us that we have a very specific way to check for divisibility by bound. But we also need to isolate a variable that's constant for each possible constraint, to make a meet in the middle attack work. So, let's separate the term with w_h into its own independent variable:

$$(2^{18} \cdot M \cdot w_h) \text{ mod } 2^{31} + ((X \cdot M + A) \gg 17) \text{ mod } 2^{31} - c_1 \cdot 2^{31} = \Delta x + k_1 \cdot \text{bound}$$

$$(2^{18} \cdot M \cdot w_h) \text{ mod } 2^{31} = c_1 \cdot 2^{31} - ((X \cdot M + A) \gg 17) \text{ mod } 2^{31} + \Delta x + k_1 \cdot \text{bound}$$

For the second equation in each constraint:

$$(2^{18} \cdot M^2 \cdot w_h) \bmod 2^{31} = c_2 \cdot 2^{31} - ((X \cdot M^2 + A(M+1)) \gg 17) \bmod 2^{31} + \Delta z + k_2 \cdot bound$$

Under mod $bound$ we get:

$$(2^{18} \cdot M \cdot w_h) \bmod 2^{31} = c_1 \cdot 2^{31} - ((X \cdot M + A) \gg 17) \bmod 2^{31} + \Delta x \pmod{bound}$$

$$(2^{18} \cdot M^2 \cdot w_h) \bmod 2^{31} = c_2 \cdot 2^{31} - ((X \cdot M^2 + A(M+1)) \gg 17) \bmod 2^{31} + \Delta z \pmod{bound}$$

where:

- $X = 2^{35} \cdot (D_h + c_0) + e_0$
- $w_h = w \gg 35$
- $c_1 = 1$ if $2^{18} \cdot M \cdot w_h + (X \cdot M + A) \gg 17 \geq 2^{31}$, $c_1 = 0$ otherwise
- $c_2 = 1$ if $2^{18} \cdot M^2 \cdot w_h + ((X \cdot M^2 + A(M+1)) \gg 17) \geq 2^{31}$, $c_2 = 0$ otherwise
- $e_0 = (w + D) \oplus M \pmod{2^{35}}$
- $e_0 = (M^{-1} \cdot (2^{17} \cdot \Delta x - A + r)) \bmod (2^{17} \cdot g) + q \cdot 2^{17} \cdot g$
 - $r \in [0, 2^{17} - 1]$
 - $q \in [0, 2^{18}/g - 1]$
 - $g = \gcd(2^{18}, bound)$

Here's a rough algorithm outline I have.

1. Bruteforce possible e_0 candidates using the following expression:

$$e_0 = (M^{-1} \cdot (2^{17} \cdot \Delta x - A + r)) \bmod (2^{17} \cdot g) + q \cdot 2^{17} \cdot g$$

- $r \in [0, 2^{17} - 1]$
- $q \in [0, 2^{18}/g - 1]$
- $g = \gcd(2^{18}, bound)$

2. Check the "weak second condition" for e_0 :

$$\Delta z - ((e_0 \cdot M^2 + A \cdot (M+1)) \gg 17) = 0 \pmod{g}$$

3. Compute w_l for each candidate e_0 . We'll use this expression:

$$e_0 = (w_l + D) \oplus M \pmod{2^{35}}$$

$$e_0 \oplus M = w_l + D \pmod{2^{35}}$$

$$w_l = e_0 \oplus M - D \pmod{2^{35}}$$

4. Compute e_0 for the rest of the structures.
5. Check both weak conditions for the rest of the structures:

$$\Delta x - ((e_0 \cdot M + A) \gg 17) = 0 \pmod{g}$$

$$\Delta z - ((e_0 \cdot M^2 + A \cdot (M+1)) \gg 17) = 0 \pmod{g}$$

6. If our candidate w_l passed all previous conditions, we'll apply our strong conditions to it. We'll use the following expressions:

$$(2^{18} \cdot M \cdot w_h) \bmod 2^{31} = c_1 \cdot 2^{31} - ((X \cdot M + A) \gg 17) \bmod 2^{31} + \Delta x \pmod{bound}$$

$$(2^{18} \cdot M^2 \cdot w_h) \bmod 2^{31} = c_2 \cdot 2^{31} - ((X \cdot M^2 + A(M + 1)) \gg 17) \bmod 2^{31} + \Delta z \pmod{bound}$$

where $X = 2^{35} \cdot (D_h + c_0) + e_0$.

So, for each w_l , we get e_0 for all structures. Then, for the first structure, we compute all possible combinations of the right hand side (bruteforcing c_0 and c_1). Remember states of c_0 and c_1 for the first structure.

For the rest of the structures, we just compute all possible combinations of the right hand side. We then find an intersection by our right hand side values of these lists.

7. Place each remaining candidate w_l in the bucket, numbered after the value of the right hand side. There should be bound buckets at most, due to us working under modulo bound
8. Once we have our buckets, try and figure out the possible values of w_h to bruteforce for each bucket from the following expression:

$$(2^{18} \cdot M \cdot w_h) \bmod 2^{31} = c_1 \cdot 2^{31} - ((X \cdot M + A) \gg 17) \bmod 2^{31} + \Delta x \pmod{bound}$$

$$2^{18} \cdot (M \cdot w_h) \bmod 2^{17} = c_1 \cdot 2^{31} - ((X \cdot M + A) \gg 17) \bmod 2^{31} + \Delta x \pmod{bound}$$