MythX

Analysis de163f06-b58a-4cca-a0e4-ff1ff464f2d9 MythX

Started	Mon Feb 19 2024 21:54:20 GMT+0000 (Coordinated Universal Time)
Finished	Mon Feb 19 2024 21:54:25 GMT+0000 (Coordinated Universal Time)
Mode	Deep
Client Tool	Mythx-Vscode-Extension
Main Source File	/Flatten/Simpleerc721ahx.Sol

DETECTED VULNERABILITIES

HIGH	MEDIUM	LOW
0	0	17

ISSUES

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
224 function average(uint256 a, uint256 b) internal pure returns (uint256) {
225 // (a + b) / 2 can overflow.
226 return (a & b) + (a ^ b) / 2;
227 }
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
224 function average(uint256 a, uint256 b) internal pure returns (uint256) {
225 // (a + b) / 2 can overflow.
226 return (a & b) + (a ^ b) / 2;
227 }
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
235 function ceilDiv(uint256 a, uint256 b) internal pure returns (uint256) {
236 // (a + b - 1) / b can overflow on addition, so we distribute.
237 return a == 0 ? 0 : (a - 1) / b + 1;
238 }
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
235 function ceilDiv(uint256 a, uint256 b) internal pure returns (uint256) {
236 // (a + b - 1) / b can overflow on addition, so we distribute.
237 return a == 0 ? 0 : (a - 1) / b + 1;
238 }
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
235 function ceilDiv(uint256 a, uint256 b) internal pure returns (uint256) {
236 // (a + b - 1) / b can overflow on addition, so we distribute.
237 return a == 0 ? 0 : (a - 1) / b + 1;
238 }
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
261 // The surrounding unchecked block does not change this fact.
262 // See https://docs.soliditylang.org/en/latest/control-structures.html#checked-or-unchecked-arithmetic.
263 return prod0 / denominator;
264 }
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
286
287 // Does not overflow because the denominator cannot be zero at this stage in the function.
288 uint256 twos = denominator & (~denominator + 1);
289 assembly {
290 // Divide denominator by twos.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
299
300 // Shift in bits from prod1 into prod0.
301 prod0 |= prod1 * twos;
302
303 // Invert denominator mod 2^256. Now that denominator is an odd number, it has an inverse modulo 2^256 such
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
304 // that denominator * inv = 1 mod 2^256. Compute the inverse by starting with a seed that is correct for
305 // four bits. That is, denominator * inv = 1 mod 2^4.
306 uint256 inverse = (3 * denominator) ^ 2;
307
308 // Use the Newton-Raphson iteration to improve the precision. Thanks to Hensel's lifting lemma, this also works
```

UNKNOWN Arithmetic operation "*=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
308 // Use the Newton-Raphson iteration to improve the precision. Thanks to Hensel's lifting lemma, this also works
309 // in modular arithmetic, doubling the correct bits in each step.
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
308 // Use the Newton-Raphson iteration to improve the precision. Thanks to Hensel's lifting lemma, this also works
309 // in modular arithmetic, doubling the correct bits in each step.
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
308 // Use the Newton-Raphson iteration to improve the precision. Thanks to Hensel's lifting lemma, this also works
309 // in modular arithmetic, doubling the correct bits in each step.
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
```

UNKNOWN Arithmetic operation "*=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
309 // in modular arithmetic, doubling the correct bits in each step.
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
309 // in modular arithmetic, doubling the correct bits in each step.
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
309 // in modular arithmetic, doubling the correct bits in each step.
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
```

UNKNOWN Arithmetic operation "*=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
310 inverse *= 2 - denominator * inverse; // inverse mod 2^8
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
```

UNKNOWN Arithmetic operation "*=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 inverse *= 2 - denominator * inverse; // inverse mod 2^256
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 inverse *= 2 - denominator * inverse; // inverse mod 2^256
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
311 inverse *= 2 - denominator * inverse; // inverse mod 2^16
312 inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 inverse *= 2 - denominator * inverse; // inverse mod 2^256
```

UNKNOWN Arithmetic operation "*=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
312 | inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 | inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 | inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 | inverse *= 2 - denominator * inverse; // inverse mod 2^256
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
312 | inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 | inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 | inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 | inverse *= 2 - denominator * inverse; // inverse mod 2^256
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
312 | inverse *= 2 - denominator * inverse; // inverse mod 2^32
313 | inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 | inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 | inverse *= 2 - denominator * inverse; // inverse mod 2^256
```

UNKNOWN Arithmetic operation "*=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
313 | inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 | inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 | inverse *= 2 - denominator * inverse; // inverse mod 2^256
316 |
317 | // Because the division is now exact we can divide by multiplying with the modular inverse of denominator.
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 inverse *= 2 - denominator * inverse; // inverse mod 2^256
316
317 // Because the division is now exact we can divide by multiplying with the modular inverse of denominator.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
313 inverse *= 2 - denominator * inverse; // inverse mod 2^64
314 inverse *= 2 - denominator * inverse; // inverse mod 2^128
315 inverse *= 2 - denominator * inverse; // inverse mod 2^256
316
317 // Because the division is now exact we can divide by multiplying with the modular inverse of denominator.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
319 // less than 2^256, this is the final result. We don't need to compute the high bits of the result and prod1
320 // is no longer required.
321 result = prod0 * inverse;
322 return result;
323 }
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
330 uint256 result = mulDiv(x, y, denominator);
331 if (rounding == Rounding.Up && mulmod(x, y, denominator) > 0) {
332 result += 1;
333 }
334 return result;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
362 // into the expected uint128 result.
363 unchecked {
364 result = (result + a / result) >> 1;
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
362 // into the expected uint128 result.
363 unchecked {
364 result = (result + a / result) >> 1;
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
363 unchecked {
364 result = (result + a / result) >> 1;
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
363 unchecked {
364 result = (result + a / result) >> 1;
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
364 result = (result + a / result) >> 1;
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
```

UNKNOWN

Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
364 result = (result + a / result) >> 1;
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
```

UNKNOWN

Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
```

UNKNOWN

Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
365 result = (result + a / result) >> 1;
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
366 result = (result + a / result) >> 1;
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
371 return min(result, a / result);
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
367 result = (result + a / result) >> 1;
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
371 return min(result, a / result);
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
371 return min(result, a / result);
372 }
```

UNKNOWN Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
368 result = (result + a / result) >> 1;
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
371 return min(result, a / result);
372 }
```

UNKNOWN

Arithmetic operation "/" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
369 result = (result + a / result) >> 1;
370 result = (result + a / result) >> 1;
371 return min(result, a / result);
372 }
373 }
```

UNKNOWN

Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
379 unchecked {
380 uint256 result = sqrt(a);
381 return result + (rounding == Rounding.Up && result * result < a ? 1 : 0);
382 }
383 }
```

UNKNOWN

Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
379 unchecked {
380 uint256 result = sqrt(a);
381 return result + (rounding == Rounding.Up && result * result < a ? 1 : 0);
382 }
383 }
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
392 if (value >> 128 > 0) {
393 value >>= 128;
394 result += 128;
395 }
396 if (value >> 64 > 0) {
397 value >>= 64;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
396 if (value >> 64 > 0) {
397 value >>= 64;
398 result += 64;
399 }
400 if (value >> 32 > 0) {
401 value >>= 32;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
400 if (value >> 32 > 0) {
401 value >>= 32;
402 result += 32;
403 }
404 if (value >> 16 > 0) {
405 value >>= 16;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
404 if (value >> 16 > 0) {
405 value >>= 16;
406 result += 16;
407 }
408 if (value >> 8 > 0) {
409 value >>= 8;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
408 if (value >> 8 > 0) {
409 value >>= 8;
410 result += 8;
411 }
412 if (value >> 4 > 0) {
413 value >>= 4;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
412 if (value >> 4 > 0) {
413 value >>= 4;
414 result += 4;
415 }
416 if (value >> 2 > 0) {
417 value >>= 2;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
416 if (value >> 2 > 0) {
417 value >>= 2;
418 result += 2;
419 }
420 if (value >> 1 > 0) {
421 result += 1;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
419 }
420 if (value >> 1 > 0) {
421 result += 1;
422 }
423 }
424 return result;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
432 unchecked {
433 uint256 result = log2(value);
434 return result + (rounding == Rounding.Up && 1 << result < value ? 1 : 0);
435 }
436 }
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
443 uint256 result = 0;
444 unchecked {
445 if (value >= 10 ** 64) {
446 value /= 10 ** 64; value /= 10 ** 64;
447 result += 64;
448 }
```

UNKNOWN Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
444 unchecked {
445 if (value >= 10 ** 64) {
446 value /= 10 ** 64;
447 result += 64;
448 }
449 if (value >= 10 ** 32) {
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
444 unchecked {
445 if (value >= 10 ** 64) {
446 value /= 10 ** 64;
447 result += 64;
448 }
449 if (value >= 10 ** 32) {
```

UNKNOWN

Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
445 if (value >= 10 ** 64) {
446 value /= 10 ** 64;
447 result += 64;
448 }
449 if (value >= 10 ** 32) {
450 value /= 10 ** 32;
```

UNKNOWN

Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
447 result += 64;
448 }
449 if (value >= 10 ** 32) {
450 value /= 10 ** 32; value /= 10 ** 32;
451 result += 32;
452 }
```

UNKNOWN

Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
448 }
449 if (value >= 10 ** 32) {
450 value /= 10 ** 32;
451 result += 32;
452 }
453 if (value >= 10 ** 16) {
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
448 }
449 if (value >= 10 ** 32) {
450 value /= 10 ** 32;
451 result += 32;
452 }
453 if (value >= 10 ** 16) {
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
449 if (value >= 10 ** 32) {
450 value /= 10 ** 32;
451 result += 32;
452 }
453 if (value >= 10 ** 16) {
454 value /= 10 ** 16;
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
451 result += 32;
452 }
453 if (value >= 10 ** 16) {
454 value /= 10 ** 16; value /= 10 ** 16;
455 result += 16;
456 }
```

UNKNOWN Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
452 }
453 if (value >= 10 ** 16) {
454 value /= 10 ** 16;
455 result += 16;
456 }
457 if (value >= 10 ** 8) {
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
452 }
453 if (value >= 10 ** 16) {
454 value /= 10 ** 16;
455 result += 16;
456 }
457 if (value >= 10 ** 8) {
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
453 if (value >= 10 ** 16) {
454 value /= 10 ** 16;
455 result += 16;
456 }
457 if (value >= 10 ** 8) {
458 value /= 10 ** 8;
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
455 result += 16;
456 }
457 if (value >= 10 ** 8) {
458 value /= 10 ** 8; value /= 10 ** 8;
459 result += 8;
460 }
```

UNKNOWN Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
456 }
457 if (value >= 10 ** 8) {
458 value /= 10 ** 8;
459 result += 8;
460 }
461 if (value >= 10 ** 4) {
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
456 }
457 if (value >= 10 ** 8) {
458 value /= 10 ** 8;
459 result += 8;
460 }
461 if (value >= 10 ** 4) {
```

UNKNOWN

Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
457 if (value >= 10 ** 8) {
458 value /= 10 ** 8;
459 result += 8;
460 }
461 if (value >= 10 ** 4) {
462 value /= 10 ** 4;
```

UNKNOWN

Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
459 result += 8;
460 }
461 if (value >= 10 ** 4) {
462 value /= 10 ** 4; value /= 10 ** 4;
463 result += 4;
464 }
```

UNKNOWN

Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
460 }
461 if (value >= 10 ** 4) {
462 value /= 10 ** 4;
463 result += 4;
464 }
465 if (value >= 10 ** 2) {
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
460 }
461 if (value >= 10 ** 4) {
462 value /= 10 ** 4;
463 result += 4;
464 }
465 if (value >= 10 ** 2) {
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
461 if (value >= 10 ** 4) {
462 value /= 10 ** 4;
463 result += 4;
464 }
465 if (value >= 10 ** 2) {
466 value /= 10 ** 2;
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
463 result += 4;
464 }
465 if (value >= 10 ** 2) {
466 value /= 10 ** 2; value /= 10 ** 2;
467 result += 2;
468 }
```

UNKNOWN Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
464 }
465 if (value >= 10 ** 2) {
466 value /= 10 ** 2;
467 result += 2;
468 }
469 if (value >= 10 ** 1) {
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
464 }
465 if (value >= 10 ** 2) {
466 value /= 10 ** 2;
467 result += 2;
468 }
469 if (value >= 10 ** 1) {
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
465 if (value >= 10 ** 2) {
466 value /= 10 ** 2;
467 result += 2;
468 }
469 if (value >= 10 ** 1) {
470 result += 1;
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
467 result += 2;
468 }
469 if (value >= 10 ** 1) {
470 result += 1; result += 1;
471 }
472 }
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
468 }
469 if (value >= 10 ** 1) {
470 result += 1;
471 }
472 }
473 return result;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
481 unchecked {
482 uint256 result = log10(value);
483 return result + (rounding == Rounding.Up && 10 ** result < value ? 1 : 0);
484 }
485 }
```

UNKNOWN Arithmetic operation "**" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
481 unchecked {
482 uint256 result = log10(value);
483 return result + (rounding == Rounding.Up && 10 ** result < value ? 1 : 0);
484 }
485 }
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
496 if (value >> 128 > 0) {
497 value >>= 128;
498 result += 16;
499 }
500 if (value >> 64 > 0) {
501 value >>= 64;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
500 if (value >> 64 > 0) {
501 value >>= 64;
502 result += 8;
503 }
504 if (value >> 32 > 0) {
505 value >>= 32;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
504 if (value >> 32 > 0) {
505 value >>= 32;
506 result += 4;
507 }
508 if (value >> 16 > 0) {
509 value >>= 16;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
508 if (value >> 16 > 0) {
509 value >>= 16;
510 result += 2;
511 }
512 if (value >> 8 > 0) {
513 result += 1;
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
511 }
512 if (value >> 8 > 0) {
513 result += 1;
514 }
515 }
516 return result;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
524 unchecked {
525 uint256 result = log256(value);
526 return result + (rounding == Rounding.Up && 1 << (result << 3) < value ? 1 : 0);
527 }
528 }
529 }
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
561 function average(int256 a, int256 b) internal pure returns (int256) {
562 // Formula from the book "Hacker's Delight"
563 int256 x = (a & b) + ((a ^ b) >> 1);
564 return x + (int256(uint256(x) >> 255) & (a ^ b));
565 }
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
562 // Formula from the book "Hacker's Delight"
563 int256 x = (a & b) + ((a ^ b) >> 1);
564 return x + (int256(uint256(x) >> 255) & (a ^ b));
565 }
566
567 /**
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
597 function toString(uint256 value) internal pure returns (string memory) {
598 unchecked {
599 uint256 length = Math.log10(value) + 1;
600 string memory buffer = new string(length);
601 uint256 ptr;
602 /// @solidity memory-safe-assembly
```

UNKNOWN Arithmetic operation "--" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
605 }
606 while (true) {
607 ptr--;
608 /// @solidity memory-safe-assembly
609 assembly {
610 mstore8(ptr, byte(mod(value, 10), _SYMBOLS))
```

UNKNOWN Arithmetic operation "/=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
610 mstore8(ptr, byte(mod(value, 10), _SYMBOLS))
611 }
612 value /= 10;
613 if (value == 0) break;
614 }
615 return buffer;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
629 function toHexString(uint256 value) internal pure returns (string memory) {
630 unchecked {
631 return toHexString(value, Math.log256(value) + 1);
632 }
633 }
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
637 */
638 function toHexString(uint256 value, uint256 length) internal pure returns (string memory) {
639 bytes memory buffer = new bytes(2 * length + 2);
640 buffer[0] = "0";
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
637 */
638 function toHexString(uint256 value, uint256 length) internal pure returns (string memory) {
639 bytes memory buffer = new bytes(2 * length + 2);
640 buffer[0] = "0";
641 buffer[1] = "x";
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
640 buffer[0] = "0";
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
643 buffer[i] = _SYMBOLS[value & 0xf];
644 value >>= 4;
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
640 buffer[0] = "0";
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
643 buffer[i] = _SYMBOLS[value & 0xf];
644 value >>= 4;
```

UNKNOWN Arithmetic operation "--" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
640 buffer[0] = "0";
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
643 buffer[i] = _SYMBOLS[value & 0xf]; buffer[i] = _SYMBOLS[value & 0xf];
644 value >>= 4;
645 }
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1418
1419 // Mask of an entry in packed address data.
1420 uint256 private constant _BITMASK_ADDRESS_DATA_ENTRY = (1 << 64) - 1;
1421
1422 // The bit position of `numberMinted` in packed address data.
1423 uint256 private constant _BITPOS_NUMBER_MINTED = 64;
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1430
1431 // Mask of all 256 bits in packed address data except the 64 bits for `aux`.
1432 uint256 private constant _BITMASK_AUX_COMPLEMENT = (1 << 192) - 1;
1433
1434 // The bit position of `startTimestamp` in packed ownership.
1435 uint256 private constant _BITPOS_START_TIMESTAMP = 160;
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1448
1449 // Mask of all 256 bits in a packed ownership except the 24 bits for `extraData`.
1450 uint256 private constant _BITMASK_EXTRA_DATA_COMPLEMENT = (1 << 232) - 1;
1451
1452 // The mask of the lower 160 bits for addresses.
1453 uint256 private constant _BITMASK_ADDRESS = (1 << 160) - 1;
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1451
1452 // The mask of the lower 160 bits for addresses.
1453 uint256 private constant _BITMASK_ADDRESS = (1 << 160) - 1;
1454
1455 // The maximum `quantity` that can be minted with {_mintERC2309}.
1456 // This limit is to prevent overflows on the address data entries.
1457 // For a limit of 5000, a total of 3.689e15 calls to {_mintERC2309}
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1545 // more than `_currentIndex - _startTokenId()` times.
1546 unchecked {
1547 return _currentIndex - _burnCounter - _startTokenId();
1548 }
1549 }
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1545 // more than `_currentIndex - _startTokenId()` times.
1546 unchecked {
1547 return _currentIndex - _burnCounter - _startTokenId();
1548 }
1549 }
```

UNKNOWN

Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1556 // and it is initialized to `_startTokenId()`.
1557 unchecked {
1558 return _currentIndex - _startTokenId();
1559 }
1560 }
```

UNKNOWN

Arithmetic operation "--" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1736 // If the address is zero, packed will be zero.
1737 while (packed == 0) {
1738 packed = _packedOwnerships[--curr];
1739 }
1740 return packed;
1741 }
```

UNKNOWN

Arithmetic operation "--" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1943 unchecked {
1944 // We can directly increment and decrement the balances.
1945 --_packedAddressData[from]; // Updates: `balance -= 1`.
1946 ++_packedAddressData[to]; // Updates: `balance += 1`.
```

UNKNOWN Arithmetic operation "++" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1944 // We can directly increment and decrement the balances.
1945 --_packedAddressData[from]; // Updates: `balance -= 1`.
1946 ++_packedAddressData[to]; // Updates: `balance += 1`.
1947
1948 // Updates:
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1958 // If the next slot may not have been initialized (i.e. `nextInitialized == false`) .
1959 if (prevOwnershipPacked & _BITMASK_NEXT_INITIALIZED == 0) {
1960 uint256 nextTokenId = tokenId + 1;
1961 // If the next slot's address is zero and not burned (i.e. packed value is zero).
1962 if (_packedOwnerships[nextTokenId] == 0) {
1963 // If the next slot is within bounds.
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2119 //
2120 // We can directly add to the `balance` and `numberMinted`.
2121 _packedAddressData[to] += quantity * ((1 << _BITPOS_NUMBER_MINTED) | 1);
2122
2123 // Updates:
2124 // - `address` to the owner.
2125 // - `startTimestamp` to the timestamp of minting.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2119 //
2120 // We can directly add to the `balance` and `numberMinted`.
2121 _packedAddressData[to] += quantity * ((1 << _BITPOS_NUMBER_MINTED) | 1);
2122
2123 // Updates:
2124 // - `address` to the owner.
2125 // - `startTimestamp` to the timestamp of minting.
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2132
2133 uint256 toMasked;
2134 uint256 end = startTokenId + quantity;
2135
2136 // Use assembly to loop and emit the `Transfer` event for gas savings.
2137 // The duplicated `log4` removes an extra check and reduces stack juggling.
2138 // The assembly, together with the surrounding Solidity code, have been
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2206 //
2207 // We can directly add to the `balance` and `numberMinted`.
2208 _packedAddressData[to] += quantity * ((1 << _BITPOS_NUMBER_MINTED) | 1);
2209
2210 // Updates:
2211 // - `address` to the owner.
2212 // - `startTimestamp` to the timestamp of minting.
```

UNKNOWN Arithmetic operation "*" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2206 //
2207 // We can directly add to the `balance` and `numberMinted`.
2208 _packedAddressData[to] += quantity * ((1 << _BITPOS_NUMBER_MINTED) | 1);
2209
2210 // Updates:
2211 // - `address` to the owner.
2212 // - `startTimestamp` to the timestamp of minting.
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2218 );
2219
2220 emit ConsecutiveTransfer(startTokenId, startTokenId + quantity - 1, address(0), to);
2221
2222 _currentIndex = startTokenId + quantity;
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2218 );
2219
2220 emit ConsecutiveTransfer(startTokenId, startTokenId + quantity - 1, address(0), to);
2221
2222 _currentIndex = startTokenId + quantity;
```

UNKNOWN

Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2220 emit ConsecutiveTransfer(startTokenId, startTokenId + quantity - 1, address(0), to);
2221
2222 _currentIndex = startTokenId + quantity;
2223 }
2224 _afterTokenTransfers(address(0), to, startTokenId, quantity);
2225 }
```

UNKNOWN

Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2248 if (to.code.length != 0) {
2249 uint256 end = _currentIndex;
2250 uint256 index = end - quantity;
2251 do {
2252 if (!_checkContractOnERC721Received(address(0), to, index++, _data)) {
2253 revert TransferToNonERC721ReceiverImplementer();
```

UNKNOWN

Arithmetic operation "++" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2250 uint256 index = end - quantity;
2251 do {
2252 if (!_checkContractOnERC721Received(address(0), to, index++, _data)) {
2253 revert TransferToNonERC721ReceiverImplementer();
2254 }
```

UNKNOWN Arithmetic operation "+=" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2321 // We can directly decrement the balance, and increment the number burned.
2322 // This is equivalent to `packed -= 1; packed += 1 << _BITPOS_NUMBER_BURNED;`.
2323 _packedAddressData[from] += (1 << _BITPOS_NUMBER_BURNED) - 1;
2324
2325 // Updates:
2326 // - `address` to the last owner.
2327 // - `startTimestamp` to the timestamp of burning.
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2321 // We can directly decrement the balance, and increment the number burned.
2322 // This is equivalent to `packed -= 1; packed += 1 << _BITPOS_NUMBER_BURNED;`.
2323 _packedAddressData[from] += (1 << _BITPOS_NUMBER_BURNED) - 1;
2324
2325 // Updates:
2326 // - `address` to the last owner.
2327 // - `startTimestamp` to the timestamp of burning.
```

UNKNOWN Arithmetic operation "+" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2335 // If the next slot may not have been initialized (i.e. `nextInitialized == false`) .
2336 if (prevOwnershipPacked & _BITMASK_NEXT_INITIALIZED == 0) {
2337 uint256 nextTokenId = tokenId + 1;
2338 // If the next slot's address is zero and not burned (i.e. packed value is zero).
2339 if (_packedOwnerships[nextTokenId] == 0) {
2340 // If the next slot is within bounds.
```

UNKNOWN Arithmetic operation "++" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2352 // Overflow not possible, as _burnCounter cannot be exceed _currentIndex times.
2353 unchecked {
2354 _burnCounter++;
2355 }
2356 }
```

UNKNOWN Arithmetic operation "++" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2541 ) external onlyRole(MINTER_ROLE) {
2542 uint256 pre = totalSupply();
2543 for (uint i = 0; i < ids.length; i++) {
2544 require(origin == ownerOf(ids[i]), "ERC721AHX: Not token owner"); require(origin == ownerOf(ids[i]), "ERC721AHX: Not token owner");
2545
2546 _burn(ids[i]);
```

UNKNOWN Arithmetic operation "-" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2548 uint256 post = totalSupply();
2549
2550 require(pre - post == ids.length, "ERC721AHX: Burning error");
2551 }
```

UNKNOWN Compiler-rewritable "<uint> - 1" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
235 function ceilDiv(uint256 a, uint256 b) internal pure returns (uint256) {
236 // (a + b - 1) / b can overflow on addition, so we distribute.
237 return a == 0 ? 0 : (a - 1) / b + 1;
238 }
```

UNKNOWN Compiler-rewritable "<uint> - 1" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2218 );
2219
2220 emit ConsecutiveTransfer(startTokenId, startTokenId + quantity - 1, address(0), to);
2221
2222 _currentIndex = startTokenId + quantity;
```

UNKNOWN Compiler-rewritable "<uint> - 1" discovered

This plugin produces issues to support false positive discovery within MythX.

SWC-101

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2321 // We can directly decrement the balance, and increment the number burned.
2322 // This is equivalent to `packed -= 1; packed += 1 << _BITPOS_NUMBER_BURNED;`.
2323 _packedAddressData[from] += (1 << _BITPOS_NUMBER_BURNED) - 1;
2324
2325 // Updates:
2326 // - `address` to the last owner.
2327 // - `startTimestamp` to the timestamp of burning.
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
8   // OpenZeppelin Contracts v4.4.1 (access/IAccessControl.sol)
9
10  pragma solidity ^0.8.0;
11
12  /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
100 // OpenZeppelin Contracts (last updated v4.9.4) (utils/Context.sol)
101
102 pragma solidity ^0.8.0;
103
104 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
132 // OpenZeppelin Contracts v4.4.1 (utils/introspection/IERC165.sol)
133
134 pragma solidity ^0.8.0;
135
136 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
161 // OpenZeppelin Contracts v4.4.1 (utils/introspection/ERC165.sol)
162
163 pragma solidity ^0.8.0;
164
165 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
192 // OpenZeppelin Contracts (last updated v4.9.0) (utils/math/Math.sol)
193
194 pragma solidity ^0.8.0;
195
196 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
535 // OpenZeppelin Contracts (last updated v4.8.0) (utils/math/SignedMath.sol)
536
537 pragma solidity ^0.8.0;
538
539 /**
540 * @dev Standard signed math utilities missing in the Solidity language.
541 */
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
582 // OpenZeppelin Contracts (last updated v4.9.0) (utils/Strings.sol)
583
584 pragma solidity ^0.8.0;
585
586
587 /**
588 * @dev String operations.
589 */
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
669 // OpenZeppelin Contracts (last updated v4.9.0) (access/AccessControl.sol)
670
671 pragma solidity ^0.8.0;
672
673
674
675
676 /**
677 * @dev Contract module that allows children to implement role-based access
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
919 // OpenZeppelin Contracts (last updated v4.9.0) (token/ERC721/IERC721.sol)
920
921 pragma solidity ^0.8.0;
922
923 /**
924 * @dev Required interface of an ERC721 compliant contract.
925 */
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1053 // OpenZeppelin Contracts v4.4.1 (interfaces/IERC721.sol)
1054
1055 pragma solidity ^0.8.0;
1056
1057
1058 // File contracts/hybridX/interfaces/IERC721HX.sol
1059
1060 // Original license: SPDX_License_Identifier: MIT
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.24"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1059
1060 // Original license: SPDX_License_Identifier: MIT
1061 pragma solidity ^0.8.24;
1062
1063 interface IERC721HX is IERC721 {
1064 function MINTER_ROLE() external returns (bytes32);
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.4"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1092 // Creator: Chiru Labs
1093
1094 pragma solidity ^0.8.4;
1095
1096 /**
1097 * @dev Interface of ERC721A.
1098 */
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.4"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
1378 // Creator: Chiru Labs
1379
1380 pragma solidity ^0.8.4;
1381
1382 /**
1383 * @dev Interface of ERC721 token receiver.
1384 */
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.4"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2471 // Creator: Chiru Labs
2472
2473 pragma solidity ^0.8.4;
2474
2475 /**
2476 * @dev Interface of ERC721ABurnable.
2477 */
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.4"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2494 // Creator: Chiru Labs
2495
2496 pragma solidity ^0.8.4;
2497
2498
2499 /**
2500 * @title ERC721ABurnable.
2501 *
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.24"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2519
2520 // Original license: SPDX_License_Identifier: MIT
2521 pragma solidity ^0.8.24;
2522
2523
2524
2525
2526 abstract contract ERC721AHX is ERC721A, ERC721ABurnable, AccessControl {
2527 string public baseURI;
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.24"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2599
2600 // Original license: SPDX_License_Identifier: MIT
2601 pragma solidity ^0.8.24;
2602
2603 contract SimpleERC721AHX is ERC721AHX {
2604 constructor(
2605 address _defaultAdmin,
```

UNKNOWN

SWC-110

Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

Source file

/flatten/simpleerc721ahx.sol

Locations

```
638 function toHexString(uint256 value, uint256 length) internal pure returns (string memory) {
639 bytes memory buffer = new bytes(2 * length + 2);
640 buffer[0] = "0";
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
```

UNKNOWN Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

SWC-110

Source file

/flatten/simpleerc721ahx.sol

Locations

```
639 bytes memory buffer = new bytes(2 * length + 2);
640 buffer[0] = "0";
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
643 buffer[i] = _SYMBOLS[value & 0xf];
```

UNKNOWN Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

SWC-110

Source file

/flatten/simpleerc721ahx.sol

Locations

```
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
643 buffer[i] = _SYMBOLS[value & 0xf];
644 value >>= 4;
645 }
```

UNKNOWN Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

SWC-110

Source file

/flatten/simpleerc721ahx.sol

Locations

```
641 buffer[1] = "x";
642 for (uint256 i = 2 * length + 1; i > 1; --i) {
643 buffer[i] = _SYMBOLS[value & 0xf];
644 value >>= 4;
645 }
646 require(value == 0, "Strings: hex length insufficient");
```

UNKNOWN Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

SWC-110

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2542 uint256 pre = totalSupply();
2543 for (uint i = 0; i < ids.length; i++) {
2544 require(origin == ownerOf(ids[i]), "ERC721AHX: Not token owner");
2545
2546 _burn(ids[i]);
```

UNKNOWN Out of bounds array access

The index access expression can cause an exception in case of use of invalid array index value.

SWC-110

Source file

/flatten/simpleerc721ahx.sol

Locations

```
2544 require(origin == ownerOf(ids[i]), "ERC721AHX: Not token owner");
2545
2546 _burn(ids[i]);
2547 }
2548 uint256 post = totalSupply();
```