19

Rick Staals and Erwin van Duijnhoven (0716184, 0719562)

2WC09- Cryptography 1

Homework sheet 2

September 20, 2013

Question 1

5/5

The idea is to find out what $h(k)$ is. When we find this, we can just add $a =$ "God@heaven.af.mil" and we are done.
Suppose we have an e-mail address a' of length n. We start by subscribing ourselves to the mailing list, after which we receive the confirmation key $h(ka')$. We then retrace n steps of the function h (subtract the final byte of the string, then rotate 4 bits to the right, the subtract the following byte of the string, etcetera). After that, we have $h(k)$. We can now calculate $h(ka)$ by adding the bytes of the string a and rotating as h does. We then have $h(ka)$, so our goal is reached.

Majordomo 1.94 should have calculated $h(ak)$ instead of $h(ka)$. That way, it is much harder to find out the secret word k or the function on the secret word $h(k)$.

yes, that solves the current problem, but you don't know whether that will introduce new weaknesses.

Question 2

4/5

We made a Matlab code that calculates this:

use accepted hash functions

```
function a = authenticator( m, r, s, p )

a=0;
mm = sscanf( sprintf( '%u', m ), '%1d' )';
n = length(mm);
rarray = zeros(n);
for j = 1:n,        n+1
    rarray(j) = mod(r^j,p);
end
for j = 1:n,
    a = mod(a + mm(j)*rarray(j),p);
end
a = mod(a + s,p);

end
```

Calling this function results in:

```
>> authenticator(45435654243597928347592843 7, 483754, 342534, 1000003)
```

```
ans =

      277544
```

So $a = 277544$.