Hello and welcome to this presentation on my dissertation project – A cryptanalysis of the GOST block cipher

Just to briefly go over what I will cover in this presentation:

I will start by providing some background to the project that will be necessary to understand the more complex details

Next, I will give a very broad overview of the project as a whole, highlighting the aims and achievements

I will then formally introduce the project deliverables before discussing them in more detail showing which ones I have completed

After this I will then present a visual representation for the related-key boomerang process which will help to explain the code demonstration that follows

Similarly, I will then show a visual representation of the sandwich process before displaying the associated code implementation

I will then finish the presentation by discussing the project conclusions and leaving time for any questions

To start this presentation, I will give a bit of background on the more complex elements of this project

So, first of all let's break down the project title.

Cryptanalysis, informally, is the art of code breaking. Specifically, it looks at the how to break ciphers using a variety of methods and techniques so as to gain access to previously inaccessible information or to give details on the security of the cipher itself.

So, what then is a block cipher. A block cipher is a form of algorithm that has seen widespread usage in cryptography due to its proficiency to encrypt data by applying a round function repeatedly. Notable block ciphers include AES and DES which have been, or still are, used as an encryption standard for most of the world.

As part of this project, I mainly focus on two block ciphers namely KASUMI and GOST. KASUMI is an adaptation of the MISTY1 block cipher created for use in the Global System for Mobile Communications. KASUMI is not widely used and is actually weaker than its predecessor MISTY1. GOST on the other hand is a suite of block ciphers used by the soviet and Russian governments as the main encryption method. However, it too is considered a weaker block cipher.

All the attacks presented in this project utilise related-keys which differ from a normal pair of cryptographic keys due to the fact that related-keys are connected by a known mathematical relationship. This relationship is usually an xor operation with a known value.

One of the types of attacks presented in this project is a boomerang attack. A boomerang attack is a form of differential attack that splits a cipher into two parts, called sub-ciphers, and utilises knowledge about the structure of the cipher to gain information about the secret keys. The related-key variant of the boomerang attack differs from the original by using one of the related-keys for each of the sub-ciphers compared to using one key for both.

The other attack presented in this project is a sandwich attack. A sandwich attack is an extension of the related-key boomerang attack that now uses three sub-ciphers instead of two. The extra sub-cipher is a one round simple Feistel switch that occurs between the other two sub-ciphers.

Although all the attacks presented in this project were originally developed as full-key recovery attacks, I elected to only implement the distinguisher version of the attacks. This is due to the reduced complexity needed for the distinguishing attacks which I hoped would not cause too much workload for this project.

Wanting to utilise both cryptanalysis and block ciphers in this project, the initial phase of this project consisted of searching for something unique that had not yet been undertaken within the cryptographic community. What I was able to find was a form of differential attack known as the sandwich attack that boasted promising cipher-breaking abilities but that had seen negligible usage since its creation. At the end of the paper in which the sandwich attack was first presented, the authors Dunkelman, Keller and Shamir suggest that a point of extension upon their research would be to try and generalise the sandwich attack for block ciphers other than KASUMI. This is because the sandwich attack was specifically designed for KASUMI and leverages algorithmic inadequacies which do not necessarily exist within other block ciphers.

As such I decided that I would use this suggested point of extension as the base of my project. In order not to give myself too much work I chose to only generalise the sandwich attack for one other block cipher, namely GOST. I chose GOST because, although a more complex block cipher than KASUMI, it is considered a weaker block cipher compared to modern standards and so should be easier to apply the sandwich attack to. So, from this I was able to construct the research question – is the sandwich attack a feasible method for breaking the GOST block cipher?

Although not mentioned by Dunkelman, Keller and Shamir in the sandwich attack paper, the generalisation of the sandwich attack could have huge implications on some of the best-known attacks for modern block ciphers. This is because the sandwich attack is built upon the framework of a related-key boomerang attack which is consistently one of the best-known attacks when applied to DES or AES. This could suggest that if the sandwich attack performs better than the related key boomerang attack and can be generalised then block ciphers such as AES could be subject to a sandwich attack that breaks the cipher more than it has been previously possibly presenting widespread security issues.

With the research question and the importance of the project established I then set about creating a list of deliverables for the project that were split into three sections, basic, intermediate and advanced. A more detailed analysis of the deliverables is provided later, but, in a nutshell the basic deliverables were to replicate the work of two papers, one paper being the original sandwich attack paper, the other paper being a theoretical presentation of a related-key boomerang attack on GOST. Both of these deliverables were accomplished although there were a number of issues that resulted in the project being very behind schedule. The intermediate deliverable was to present the theory showing whether the sandwich attack could be used on GOST. This was also achieved as I was able to show that the sandwich attack on GOST would have at most the same complexity as a related-key boomerang attack. Hence, I was able to answer the research question – yes, the sandwich attack is a feasible method for breaking the GOST block cipher. Finally, I was unable to fully complete either of the advanced deliverables due to the project being behind schedule however an attempt at simplifying an explanation of the sandwich attack process is given as part of this presentation.

I will now go into more detail about the deliverables for this project.

The first basic deliverable was to replicate the sandwich distinguisher on KASUMI presented by Dunkelman, Keller and Shamir. The reasoning behind this was to show that

1. The sandwich attack was a method that worked in practice and
2. That the results of the original paper were repeatable

By doing this I hoped it would also provide me with a better understanding of how the sandwich distinguisher works so that when it came to doing the later deliverables I would fair better

The second basic deliverable was to create the first known implementation and experimental verification of the theoretical related-key boomerang distinguisher on GOST presented by Rudksoy. The reasoning behind this being that I would be able to show the first verification of this attack and confirm whether the theory presented holds in practice. Furthermore, it would also give me a basis from which to construct the sandwich attack on GOST.

The intermediate deliverable was essentially to answer the research question. I would do this by showing the theory behind why the sandwich attack would or would not work when applied to GOST.

Finally, the advanced deliverables were dependent on the outcome of the intermediate deliverable. If I found that the sandwich distinguisher was feasible on GOST then I would go on to create an implementation of the distinguisher to try and experimentally verify my expected results. Alternatively, if I found that it was infeasible then I would go on to add to the documentation of the sandwich attack. This is because there is very little information that exists surrounding the sandwich attack and the information that does exist is quite complex, hence, I wanted to provide materials that would hopefully help the wider cryptographic community understand the sandwich attack in better detail.

For the first basic deliverable the initial plan was to create my own version of the KASUMI block cipher however…

After implementing KASUMI I then experienced issues implementing the sandwich attack. This turned out to be because I was using a version of the sandwich attack paper released in 2010 which had not been moderated and contained lots of errors. However, I was able to complete the implementation after finding a moderated version of the paper released in 2014 despite the fact that this paper also contained errors.

With the implementation of the distinguisher working, I was then able to achieve the results as seen in figure 1 which despite the low number of trails clearly shows values suitably close to the theory to suggest the implementation and subsequently the paper were correct

Despite this I was unable to make a direct comparison to the results achieved in the sandwich attack paper as Dunkelman, Keller and Shamir utilise a slight improvement of the first differential, which results in the probability of a right quartet occurring increase by a factor of two, which I did not implement.

Due to the issues faced with the first basic deliverable I elected not to try and implement an original version of GOST as was initially planned and instead used an existing implementation so as to save time and help get the project back on track. I did however use a number of Known Answer Tests in order to ensure that this version of GOST was correct.

The main part of the second basic deliverable was to create the first implementation of and subsequently the first experimental verification of the theoretical related-key boomerang attack presented in Rudskoy’s paper. It is worth noting that the attack presented by Rudskoy was first presented by Fleischmann et al however I was unable to locate the paper by Fleischmann et al. Furthermore, Rudskoy goes on to say that the attack presented by Fleischmann et al is wrong and is in fact no better than exhaustive search. However, Rudskoy does not go on to prove those claims experimentally.

As was expected I encountered some issues when implementing the related-key boomerang attack. Notably I was unable to obtain any right quartets. It was later found out that this was due to one of the differentials presented by Rudskoy not working within my implementation. Hence, I then had to perform some testing to find new differentials that would work with my implementation. I was able to find some new differentials however from my analysis the differentials in questions appeared to occur with significantly increased probability. Specifically, my differentials were occurring with probability ¼ where as Rudskoy’s theory suggested his should appear with probability 2^-6.

However, with the distinguisher now working I was able to produce the results in figure 2. From the graph it is clear to see that a Poisson distribution with a mean of 16 fits the data very well. However, this mean value of 16 suggests that the probability of a right quartet occurring was 2^-4 and not 2^-2 like I had predicted. I am still unsure as to why this value is different and would suggest more rigorous analysis be performed.

Despite this I believe I was able to successfully create a related-key boomerang distinguisher albeit if I was unable to verify its validity

The advanced deliverable for this project required me to perform some analysis on the GOST block cipher to establish whether the sandwich attack could be used on it. The first step was to figure out where to put the filling layer. I was able to work out that due to the key schedule of GOST being inverted from round 24 to round 25, the applied key difference at round 24 was essentially undone on round 25. What this meant was that I would be able to perform a simple Feistel switch on round 25 and obtain known output differentials with a probability of 1. This allowed me to then say that the filling layer of the sandwich attack could be added at round 25 and subsequently the sandwich attack was possible to perform on GOST. Furthermore, because the filling layer appears with probability 1, I was also able to say that the complexity of the sandwich attack on GOST would be no worse than the complexity of the related-key boomerang attack on GOST hence showcasing the possible efficiency of the attack. Combining all of this allowed me to answer the research question because if the sandwich attack on GOST is no worse than the related-key boomerang attack on GOST then because the related-key boomerang attack is a feasible method of attack in practice then so must be the sandwich attack.

Due to all the issues that caused significant delays within this project, unfortunately I was unable to fully complete any of the advanced deliverables. I have however created a simplified visualisation of the sandwich process which I will present in this presentation. This partially completes the second of the advanced deliverables however because I am yet to publish this visualisation, I cannot say that I have completed it fully.

Much like Dunkelman, Keller and Shamir, in order to describe the sandwich distinguisher process, I first start by presenting a visual representation for the workings of the related-key boomerang distinguisher

The following visual demonstration shows the process of the sandwich distinguisher. It is very similar to the related-key boomerang distinguisher process but with a few notable differences.

To conclude this presentation, I will review the project conclusions.

Firstly, I was able to successfully implement the sandwich distinguisher on KASUMI and replicate results from Dunkelman, Keller and Shamir. Although my implementation was slightly different, it still followed the theory presented in the sandwich attack paper.

Next, I was able to implement a previously unimplemented theoretical related-key boomerang distinguisher on GOST and provide the first experimental verification of said distinguisher. Despite the experimental results suggesting that the theory behind the distinguisher was wrong I was still able to generate right quartets after implementing a new differential. The experimental results however also did not line up with the theory I suggested and so I was unable to validate the claims I had made.

Then, I went on to show that the sandwich distinguisher could be applied to the GOST block cipher which suggested that the sandwich attack was in fact a feasible method for breaking the GOST block cipher, hence, answering the research question.

Although I was unable to create an implementation of the sandwich distinguisher on GOST and validate the claims made in this project, I was able to make a start on generating some better documentation of the sandwich attack notably creating a simplified description of how it works.